

A data processing system which permits user dialogue in a plurality of natural languages

Veröffentlichungsnr. (Sek.) DE4325096
Veröffentlichungsdatum : 1995-02-02
Erfinder : CHARLES JACQUES (FR); MERZ LAUREN (US)
Anmelder : COMPUTERVISION CORP (US)
Veröffentlichungsnummer : ☐ DE4325096

Aktenzeichen:
(EPIDOS-INPADOC-normiert) DE19934325096 19930727
Prioritätsaktenzeichen:
(EPIDOS-INPADOC-normiert) DE19934325096 19930727
Klassifikationssymbol (IPC) : G06F3/16 ; G10L3/00 ; G06F17/28
Klassifikationssymbol (EC) : G06F3/16, G06F9/44W6, G10L15/26C
Korrespondierende Patentschriften

Bibliographische Daten

A data processing system which permits user dialogue in a plurality of natural languages contains a processor for executing application programs, a memory for storing message (information, communication) files, which are associated with the application program, in at least one natural language and a link table for storing links between the application program and associated message files, a link storing an application program of one natural language and a pointer which indicates its message files. The operating system of the processor constructs links when a message file is loaded in the memory. The data processing system nests language information for nested application programs. A recovery mechanism is used to provide a standard language when a language desired by the user is not available and/or when nested languages are not available. The system can send and receive messages in a plurality of natural languages with the simultaneous use of different application programs in a manner which is clear to the user and the application programmer with support from the involvement of the operating system in the construction of links with the use of standard languages and the nesting of language

information.



Daten aus der esp@cenet Datenbank - - 12



DEUTSCHES
PATENTAMT

21 Aktenzeichen: P 43 25 096.3
22 Anmeldetag: 27. 7. 93
43 Offenlegungstag: 2. 2. 95

71 Anmelder:
Computervision Corp., Bedford, Mass., US

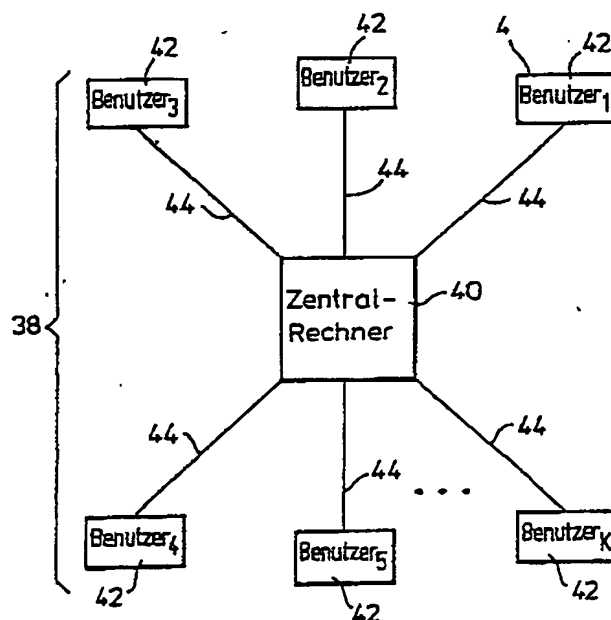
74 Vertreter:
Köhler, R., Dipl.-Phys.; Schmid, B., Dipl.-Ing.;
Holzmüller, R., Dipl.-Ing. Dr.-Ing.; Rüdel, D.,
Dipl.-Ing. Dipl.-Wirtsch.-Ing.; Bähring, A., Dipl.-Phys.
Univ. Dr.rer.nat., Pat.-Anwälte, 70565 Stuttgart

72 Erfinder:
Merz, Lauren, Newton, Mass., US; Charles, Jacques,
Chartrettes, FR

Prüfungsantrag gem. § 44 PatG ist gestellt

54 Ein Datenverarbeitungssystem, welches einen Benutzer-Dialog in mehreren, natürlichen Sprachen ermöglicht

57 Ein Datenverarbeitungssystem, welches einen Benutzer-Dialog in mehreren natürlichen Sprachen ermöglicht, enthält einen Prozessor zur Ausführung von Anwendungsprogrammen und einen Speicher zur Speicherung von dem Anwendungsprogramm zugehörigen Nachrichtendateien in wenigstens einer natürlichen Sprache und eine Verbindungstabelle zur Speicherung von Verbindungen zwischen dem Anwendungsprogramm und zugehörigen Nachrichtendateien, wobei eine Verbindung eines Anwendungsprogramms einer natürlichen Sprache und einen seine Nachrichtendateien anzeigenden Zeiger speichert, wobei das Betriebssystem des Prozessors Verbindungen aufbaut, wenn eine Nachrichtendatei im Speicher geladen ist. Das Datenverarbeitungssystem verschachtelt Sprachinformationen für verschachtelte Anwendungsprogramme. Ein Rückzugsmechanismus wird verwendet, um eine Standardsprache bereitzustellen, wenn eine durch den Benutzer gewünschte Sprache nicht verfügbar ist und oder wenn verschachtelte Sprachen nicht verfügbar sind. Das System kann Nachrichten in mehreren natürlichen Sprachen unter simultaner Verwendung verschiedener Anwendungsprogramme in einer Weise senden und empfangen, die dem Benutzer und dem Anwendungsprogramm mit Unterstützung der Betriebssystembeteiligung an dem Aufbau von Verbindungen unter Verwendung von Standardsprachen und der Verschachtelung von Sprachinformationen ersichtlich ist.



Beschreibung

Gebiet der Erfindung

Die Erfindung betrifft Datenverarbeitungssysteme im allgemeinen und insbesondere Datenverarbeitungssysteme, welche mit dem Benutzer in mehreren, natürlichen Sprachen kommunizieren können.

Hintergrund der Erfindung

Die meisten Datenverarbeitungs- und Computersysteme ermöglichen einem Benutzer den Dialog nur in einer natürlichen Sprache, wie englisch, französisch oder deutsch. Derartige natürliche Sprachen sind solche, die Personen zum Sprechen verwenden und unterscheiden sich von Computer- oder Programmiersprachen. Aufgrund der internationalen Entwicklung, in der immer häufiger Computer verwendet werden, ist ein Bedarf an Datenverarbeitungssystemen entstanden, die mit dem Benutzer in einer von mehreren natürlichen Sprachen, den "Dialog-Sprachen" kommunizieren.

In den meisten Datenverarbeitungssystemen beinhaltet der Dialog mit einem Benutzer das Senden und Empfangen von Nachrichten zwischen dem Benutzer und dem Datenverarbeitungssystem. Um solch ein Senden und Empfangen von Nachrichten durchzuführen hat ein Datenverarbeitungssystem normalerweise Standardbefehle zum Senden von Ausgabenachrichten zum Benutzer und ebenso Standardbefehle zum Übersetzen der vom Benutzer empfangenen Nachrichten. Ein Anwendungsprogrammierer verwendet diese Befehle in einem Anwendungsprogramm um einen Dialog zwischen einem Benutzer und dem Datenverarbeitungssystem zu ermöglichen, wenn das Programm läuft, d. h. wenn das System durch das Programm konfiguriert ist und durch das Programm beschriebene Verfahren durch das konfigurierte System ausgeführt werden.

Eine bekannte Technik zum Senden und Empfangen von Nachrichten beinhaltet Daten von Ausgabe- und Eingabenachrichten für ein Anwendungsprogramm in einer Datei oder an einer vom Anwendungsprogramm getrennten Stelle. Solche Nachrichten werden durch das Anwendungsprogramm mit Codes oder Indizes versehen. Diese Technik ist als "Veräußerlichung" bekannt. Gelegentlich erhalten die Anwendungsprogramme Nachrichten, d. h. wenn sie nicht veräußerlicht sind, beispielsweise zur Verwendung als Standardnachrichten. Manchmal werden diese Nachrichten auch als "Hart-kodiert" bezeichnet.

Wenn ein Anwendungsprogramm und dazu korrespondierende, veräußerlichte Nachrichtendateien in einem System eingesetzt werden, dann wird während der Laufzeit eine Verbindung zwischen dem Anwendungsprogramm und der Nachrichtendatei geschaffen. Diese Verbindung spezifiziert, daß die Nachrichtendatei in Verbindung mit dem Anwendungsprogramm zu verwenden ist um Ausgabe und Eingabenachrichten für das Programm bereit zu stellen. In gegenwärtigen Systemen muß entweder der Benutzer oder der Anwendungsprogrammierer die Nachrichtendatei, die mit dem Anwendungsprogramm verwendet werden soll, explizit spezifizieren. Somit ist die Verbindung für den Programmierer nicht ersichtlich und sowohl der Anwendungsprogrammierer als auch der Benutzer muß die Namen von verfügbaren Nachrichtendateien kennen. In einem System, welches mehrere Dialog-Sprachen unterstützt, werden

mehrere Nachrichtendateien benötigt, so daß die auf dem Programmierer und dem Benutzer liegende Last vergrößert wird.

Ein gegenwärtiges, mehrsprachiges System, welches auf einer "Veräußerung" beruht, ist das Native Language System (NLS). NLS ist eine von X/OPEN entwickelte Systemschnittstelle. X/OPEN ist eine Gemeinschaftsinitiative von Mitgliedern der Wirtschaftsgemeinschaft um entwickelte Standards in einer gemeinsamen, vorteilhaften und weiterführenden Strategie zu integrieren. Die Systemschnittstelle ist eine Gruppe von Befehlen, die zur Verwendung in Anwendungsprogrammen erhältlich sind um ein Datenverarbeitungssystem zu konfigurieren und zu manipulieren. Es existieren viele Verwirklichungen zur Unterstützung der NLS-Schnittstelle, aber alle Verwirklichungen beruhen auf der X/OPEN in ihren Portability guide, 1989, (siehe besonders Band 2) definierten Funktionalität.

Im NLS ist jede veräußerlichte Nachrichtendatei eines Anwendungsprogrammes, die Ausgabe- und Eingabenachrichten in einer Dialogsprache enthält, gekennzeichnet durch einen als "Katalog-Descriptor" bekannter verketteter Namen. Mit dem NLS muß ein Anwendungsprogrammierer den Katalog-Descriptor für eine Nachrichtendatei kennen, um diese Nachrichtendatei in den Speicher zu laden und um Nachrichten daraus abzurufen. Der Katalog-Descriptor kann aus einem Standardwort, wie einem systemweiten Standard oder einem pro-Benutzer-Standard gebildet sein. Im NLS überschreibt das Benutzer-Standardwort den Systemstandard und kann auch benutzt werden um den Standard für ein Anwendungsprogramm zu überschreiben. Eine Fähigkeit zur Verschachtelung ist nicht geschaffen. Wenn eine Nachrichtendatei in einer gewünschten Sprache für ein Anwendungsprogramm nicht verfügbar ist, dann erhält der Benutzer normalerweise hart-kodierte Nachrichten des Anwendungsprogrammes. Weil der Anwendungsprogrammierer den Namen der Nachrichtendatei in dem Anwendungsprogramm verwendet und weil er auch das Öffnen und Schließen von Nachrichtendateien durchführt, entstehen Probleme in der Entwicklung von Anwendungsprogrammen.

Ein erstes Problem mit dem NLS-System besteht in der erforderlichen Beteiligung des Anwendungsprogrammierers in der Handhabung von Nachrichtendateien. Nachrichtendateien für ein Anwendungsprogramm werden regelmäßig modifiziert oder neue Nachrichtendateien werden zur Verwendung durch das Anwendungsprogramm hinzugefügt. Derartige Modifikationen und Hinzufügungen können Änderungen in Anwendungsprogrammen, die die modifizierten oder hinzugefügten Nachrichtendateien betreffen, erforderlich machen. Daraus ergibt sich, daß wenn solche Nachrichtendateien einem Kunden zur Verfügung gestellt werden, daß auch das Anwendungsprogramm aktualisiert werden muß, so daß sowohl die Herstellungskosten des Anwendungsprogramms als auch die Materialkosten des Kunden steigen.

Eine zweite Schwierigkeit des NLS-Systemes ist, daß es einen Anwendungsprogrammierer erfordert, um ein Programm zu Handhabungszwecken zu entwickeln für den Fall, daß eine Nachrichtendatei während des Versuches, eine Datei zu öffnen oder ein Nachricht aus einer Datei abzurufen, nicht gefunden wird. Solch ein Programm stellt typischerweise eine hart-kodierte Standardnachricht zur Verwendung bereit, wenn eine Nachricht oder eine Nachrichtendatei nicht gefunden wird. Jedoch werden derartige Standardnachrichten lediglich

dem Anwendungsprogrammierer bereitgestellt. Somit können Anwendungsprogramme, die durch verschiedene Anwendungsprogrammierer hergestellt wurden, unter Umständen keinen einheitlichen Standardmechanismus aufweisen. Daher entsprechen solche Anwendungsprogramme nicht den diesbezüglichen Anforderungen.

Mehrsprachige Dialoge im NLS sind also für den Benutzer nicht ersichtlich. Beispielsweise kann der Benutzer aufgefordert werden, eine Dialogsprache auszuwählen. Jedoch wird diese Fähigkeit nur bei einem umsichtigen Anwendungsprogrammierer bereitgestellt, so daß darauf kein Verlaß ist. Dieser Mangel eines verlässlichen Mechanismus zur Auswahl einer Dialogsprache kann sowohl für den Benutzer als auch für den Anwendungsprogrammierer verwirrend sein. Darüberhinaus muß ein Benutzer, wenn er ein geschichtetes Verfahren verwendet, oft eine gewünschte Dialogsprache mehr als einmal spezifizieren. Solche unverlässlichen Systeme sind unbequem und langweilig zu benutzen und zu erhalten.

Schließlich sind diese Systeme, teilweise aufgrund des Mangels an Durchsichtigkeit und Verlässlichkeit, nicht in der Lage von der "Verschachtelungs"-Fähigkeit eines Systems Gebrauch zu machen. D.h. wenn ein Anwendungsprogramm ein anderes Anwendungsprogramm betreibt (d. h. es macht einen "verschachtelten" Aufruf), dann wird die Information über die Dialogsprache nicht "verschachtelt". Ein Benutzer ist also nicht in der Lage verschiedene Sprachen für verschiedene Programmebenen zu spezifizieren.

Anwendungswörterbücher werden üblicherweise in großen Mehrfach-Benutzer-Systemen verwendet und enthalten übliche Unterprogramme, auf die eine Vielzahl von Anwendungsprogrammen zugreifen. Der Zugriff auf diese Anwendungswörterbücher beinhaltet oft eine Art von verschachteltem Aufruf. Ohne die Schaffung einer mehrsprachigen Unterstützung für verschachtelte Anwendungsprogramme oder Wörterbücher kann die Nachrichtenhandhabung für Anwendungswörterbücher unverlässlich oder sogar unvorhersehbar sein.

Daher liegt der Erfindung die Aufgabe zugrunde, ein Datenverarbeitungssystem und ein Verfahren zu schaffen, welches in der Lage ist, den Bediener-Dialog in jeder von einer Vielzahl von natürlichen Sprachen in einer Weise zu unterstützen, die sowohl dem Bediener als auch dem Anwendungsprogrammierer ersichtlich ist.

Eine weitere Aufgabe der Erfindung besteht darin, ein Datenverarbeitungssystem zu schaffen, in welchem die Verwaltung der Nachrichtendateien für den Anwendungsprogrammierer ersichtlich ist.

Eine weitere Aufgabe der Erfindung besteht darin, ein Datenverarbeitungssystem zu schaffen, welches eine Nachrichtendatei mit einem Anwendungsprogramm oder mit einer Bibliothek ohne einen Bediener-Dialog verbindet.

Es ist noch eine Aufgabe der Erfindung, ein Datenverarbeitungssystem zu schaffen, welches verschachtelte Aufrufe in verschiedenen Dialogsprachen unterstützt.

Zusammenfassung der Erfindung

Um die oben genannten und andere Aufgaben, Merkmale und Vorteile der Erfindung zu erreichen, ist ein Datenverarbeitungssystem geschaffen worden, welches Anwendungsprogramme ausführt und welches den Dialog mit einem Benutzer in mehreren, natürlichen Sprachen durchführt. Ein Benutzer kann mit einem Pro-

gramm in einer Sprache und mit einem anderen Programm in einer anderen Sprache kommunizieren. Andere Benutzer können mit demselben Programm in noch weiteren, verschiedenen Sprachen kommunizieren. Jedes Anwendungsprogramm hat wenigstens eine korrespondierende Nachrichtendatei, die Nachrichten in einer natürlichen Sprache enthält. Das Datenverarbeitungssystem speichert Anwendungsprogramme und korrespondierende Nachrichtendateien und führt die Verarbeitungsschritte der gespeicherten Anwendungsprogramme aus. Ein Anwendungsprogramm kann Nachrichten von einem seiner korrespondierenden Nachrichtendateien in einer ausgewählten natürlichen Sprache abrufen. Um die Nachricht abzurufen speichert das Datenverarbeitungssystem Verbindungen in einer Verbindungstabelle, in welcher jede Verbindung eines Anwendungsprogrammes oder eine Wörterbuches die Lage der korrespondierenden Nachrichtendatei bereithält, die die Nachrichten in der für das Anwendungsprogramm oder das Wörterbuch gegebenen natürlichen Sprache enthält. Für ein Anwendungsprogramm oder ein Wörterbuch können in dem Datenverarbeitungssystem nebeneinander mehrere natürliche Sprachen, vorzugsweise durch das Betriebssystem, bereit gehalten werden. Eine Verbindung in der Verbindungstabelle wird vorzugsweise durch den Prozeß des Ladens einer Nachrichtendatei eines Anwendungsprogrammes in den Speicher erstellt. Die Verbindungstabelle wird abgesucht, um die Nachrichtendatei für eine ausgewählte, natürliche Sprache herauszufinden und anschließend wird die gewünschte Nachricht in der durch die Verbindung angezeigten Nachrichtendatei gefunden. Die ausgewählte, natürliche Sprache wird zumindest aus der momentan verwendeten Dialogsprache eines gegebenen Anwendungsprogrammes und einer Standardsprache ausgewählt. Das System speichert vorzugsweise sowohl die momentane Dialogsprache als auch die Standardsprache. Üblicherweise wird die Standardsprache initialisiert, wenn ein Benutzer beginnt, das Datenverarbeitungssystem zu benutzen. Die momentane Dialogsprache wird bestimmt, wenn ein Anwendungsprogramm aktiviert wird.

Bei der Abrufung von Nachrichten wird eine Dialogsprache ausgewählt aus zumindest einer momentanen Dialogsprache und der Standarddialogsprache. Dann wird eine Nachricht unter Verwendung der ausgewählten Dialogsprache aufgerufen.

Sobald das Datenverarbeitungssystem angewiesen worden ist, ein Anwendungsprogramm auszuführen, wird die momentane Dialogsprache in einer dem Anwendungsprogramm und dem Benutzer ersichtlichen Weise initialisiert. Die momentane Dialogsprache wird normalerweise als Standardsprache bestimmt; jedoch kann eine gewünschte Dialogsprache bestimmt werden, wenn das Programm aufgerufen ist. Sobald eine Dialogsprache bestimmt ist, wird die momentane Dialogsprache als die gewünschte Dialogsprache gesetzt, falls sie verfügbar ist.

Weiterhin speichert das System vorzugsweise verschachtelte Sprachinformationen zur Verwendung in Verbindung mit verschachtelten Anwendungsprogrammen oder Geschichteten Verfahrensschritten. Wenn ein Anwendungsprogramm ein anderes Anwendungsprogramm oder ein Wörterbuch aufruft, dann wird die momentane Dialogsprache als die verschachtelte Sprachinformation abgespeichert. Diese verschachtelte Sprachinformation ist vorzugsweise ein Stapelregister oder eine verbundene Liste. Dann wird die momentane

Dialogsprache als zumindest eine der folgenden Sprachen gesetzt: eine angezeigte, gewünschte Dialogsprache, die vorhergehende momentane Dialogsprache oder eine Standardsprache. Wenn ein geschachteltes Anwendungsprogramm beendet ist, wird das vorhergehende Anwendungsprogramm wieder abgespeichert und für das vorhergehende Anwendungsprogramm wird die momentane Dialogsprache zurückgesetzt zu der vorhergehenden Dialogsprache. Die vorhergehende Dialogsprache und die verschachtelte Sprachinformation für das wieder abgespeicherte Anwendungsprogramm kann dann von den Mitteln zur Speicherung verschachtelter Sprachinformationen zurückgeholt werden.

Durch die Verwendung entweder von Standardsprachen oder von der Fähigkeit, verschachtelte Sprachen zu vermeiden, werden Anwendungswörterbücher mit mehrsprachigen Eingabe und Ausgabefähigkeiten versehen. Unterprogrammaufrufe an Anwendungswörterbüchern können in ähnlicher Weise wie verschachtelte Anwendungsprogramme behandelt werden.

In der bevorzugten Ausführungsform der Erfindung verarbeitet das Betriebssystem mehrsprachige Dialoge. Ein Anwendungsprogramm instruiert das Betriebssystem eine Nachrichtendatei zu laden und Nachrichten davon aufzurufen. Die Durchführung dieser Prozesse durch das Betriebssystem machte Bedienungsfehler, Sprachverschachtelungen und Standardmechanismen ersichtlich und verlässlich.

Kurze Beschreibung der Zeichnungen

Die Bedienung und die Vorteile der Erfindung werden durch die folgende ausführliche Beschreibung besser zu verstehen sein, die in Verbindung mit der beigefügten Zeichnung zu lesen sind, in welcher:

Fig. 1 ein Blockdiagramm eines zur Durchführung der Erfindung geeigneten Datenverarbeitungssystems zeigt;

Fig. 2 eine dreidimensionale Ansicht einer typischen Benutzerstation des Datenverarbeitungssystems gemäß Fig. 1 zeigt;

Fig. 3 ein Blockdiagramm die Struktur des Zentralrechners gemäß Fig. 1 zeigt;

Fig. 4 eine schematische Darstellung der Aufteilung des Speichers gemäß Fig. 3 in private und gemeinsame Abschnitte zeigt;

Fig. 5 ein Diagramm einer Datenstruktur einer Verbindungstabelle zur Speicherung von Verbindungen zwischen Ausgabe-Nachrichtendateien und Anwendungsprogrammen und Wörterbüchern zeigt;

Fig. 6 ein Diagramm einer Datenstruktur einer System-Informationen-Tabelle zur Speicherung von Informationen über den momentanen Status der Dialogkapazitäten zeigt;

Fig. 7 ein Diagramm einer Datenstruktur einer System-Sprach-Tabelle zur Speicherung von Sprachnamen und korrespondierender Sprachcodes zeigt;

Fig. 8 ein Diagramm einer Datenstruktur einer System-Freigabe-Tabelle zur Speicherung von Informationen über die Freigabe der Dialog-Kapazitäten zeigt;

Fig. 9 ein Diagramm einer Datenstruktur zur Speicherung der Sprachcode für die Benutzer-Standardsprache zeigt;

Fig. 10 ein Diagramm einer Datenstruktur eines durch den Benutzer verschlüsselten Sprach-Stapelregisters zur Speicherung von Dialog-Sprachinformationen für verschachtelte Programme zeigt;

Fig. 11 ein Diagramm einer Datenstruktur einer Be-

nutzer-Sprach-Tabelle zur Speicherung von Informationen über die von einem speziellen Benutzer benutzten Dialog-Sprachen zeigt;

Fig. 12 ein Diagramm einer Datenstruktur einer Benutzer-Sprach-Korrespondenztabelle zur Speicherung von Dialog-Sprachnamen und Codes eines Benutzers und korrespondierenden Systemcodes zeigt;

Fig. 13 ein Diagramm einer Datenstruktur einer Benutzer-Informationen-Tabelle zur Speicherung von Informationen über die momentanen Dialogkapazitäten eines Benutzers zeigt;

Fig. 14 ein Fließdiagramm zeigt, welches beschreibt, wie das Datenverarbeitungssystem gemäß Fig. 1 konfiguriert werden kann, um vollständig freigegeben zu werden, um in mehreren Sprachen zu kommunizieren;

Fig. 15 ein Fließdiagramm zeigt, welches beschreibt, wie das Datenverarbeitungssystem gemäß Fig. 1 konfiguriert werden kann, um in einer vorgegebenen Anzahl von Sprachen zu kommunizieren;

Fig. 16 ein Fließdiagramm zeigt, welches beschreibt, wie eine Betriebssystem-Standardsprache geändert werden kann;

Fig. 17 ein Fließdiagramm zeigt, welches beschreibt, wie eine Nachrichtendatei einer Dialog-Sprache für ein gemeinsames Anwendungsprogramm geladen werden kann;

Fig. 18 ein Fließdiagramm zeigt, welches beschreibt, wie ein gemeinsames Anwendungsprogramm in den Speicher geladen werden kann;

Fig. 19 ein Fließdiagramm zeigt, welches beschreibt, wie das Datenverarbeitungssystem konfiguriert werden kann und wie es privat zum Dialog in mehreren Sprachen für ausschließlich private Anwendungsprogramme freigegeben werden kann;

Fig. 20 ein Fließdiagramm zeigt, welches beschreibt, wie mehrsprachige Dialoge für einen Benutzer über eine erste Verwendung von dem Datenverarbeitungssystem initialisiert werden kann;

Fig. 21 ein Fließdiagramm zeigt, welches beschreibt, wie die Benutzer-Standardsprache geändert werden kann;

Fig. 22 ein Fließdiagramm zeigt, welches beschreibt, wie ein Anwendungsprogramm für einen mehrsprachigen Gebrauch aufgerufen werden kann;

Fig. 23A bis Fig. 23B zeigen Fließdiagramme, die die Schritte zur Auffindung einer Nachrichtendatei und die Ladung der Nachrichtendatei in den Speicher zeigt;

Fig. 24 ein Fließdiagramm zeigt, welches beschreibt, wie eine Nachrichtendatei eines Anwendungsprogrammes in einen Pro-Benutzer-Speicher überführt werden kann;

Fig. 25 ein Fließdiagramm zeigt, welches beschreibt, wie eine Anwender-Sprach-Tabelle erweitert werden kann;

Fig. 26 ein Fließdiagramm zeigt, welches die Schritte zur Auffindung einer Nachricht für eine Standard-Aufgabe-Funktion zeigt;

Fig. 27 ein Fließdiagramm zeigt, welches die Schritte zur Auffindung einer Nachricht für ein gemeinsames Anwendungsprogramm zeigt;

Fig. 28 ein Fließdiagramm zeigt, welches die Schritte zur Auffindung einer Nachricht für ein privates Anwendungsprogramm zeigt;

Fig. 29 ein Fließdiagramm zeigt, welches beschreibt, wie eine Eingabe eines Benutzers in einer natürlichen Sprache durch das Datenverarbeitungssystem verstanden werden kann;

Fig. 30 ein Fließdiagramm zeigt, welches beschreibt, wie ein Anwendungsprogramm oder ein Wörterbuch und seine Nachrichtendateien aus dem Speicher oder dem Gebrauch zurückgebracht werden können;

Fig. 31 ein Fließdiagramm zeigt, welches beschreibt, wie eine Anwendungsprogramm-Ebene nach der Beendigung eines verschachtelten Aufrufes wieder abgespeichert werden kann;

Fig. 32 ein Fließdiagramm zeigt, welches beschreibt, wie eine Liste von bestehenden Dialog-Sprachen eines Anwendungsprogrammes bereitgestellt werden kann; und

Fig. 33 ein Fließdiagramm zeigt, welches beschreibt, wie eine Liste von verfügbaren Dialog-Sprachen bereitgestellt werden kann.

Ausführliche Beschreibung der Erfindung

Die Fig. 1 bis 4 stellen die Struktur eines Datenverarbeitungssystems 38 zur Durchführung der Erfindung dar. Das System 38 enthält wenigstens einen Zentralrechner 40, der mit wenigstens einer Benutzerstation 42 durch eine Kommunikationsverbindung 44, die einem Kabel oder einem Kommunikationsbus, wechselseitig verbunden ist. Zusätzlich können der Zentralrechner 40 und die Benutzerstationen 42 wechselseitig mit dem System in einer ähnlichen Weise verbunden sein. Fig. 1 zeigt K mit dem Zentralrechner verbundene Benutzerstationen 42. Jede Benutzerstation ist als Benutzer mit Index n gekennzeichnet, wobei $n = 1 \dots K$ sein kann. Jede Benutzerstation 42 erlaubt es einem Benutzer, mit dem Zentralrechner 40 wechselseitig zu kommunizieren; so daß mehrere Benutzerstationen 42 es mehr als einem Benutzer ermöglichen, das System gleichzeitig zu benutzen. Ebenso kann mehr als ein Benutzer eine Benutzerstation verwenden. Die Benutzerstation 42 (Fig. 2) kann eine Tastatur 46, einen Monitor 48 und andere Vorrichtungen enthalten, die den Benutzer in die Lage versetzen, Nachrichten in einer natürlichen Sprache zu empfangen und zu übermitteln.

Wie in Fig. 3 gezeigt ist, kann der Zentralrechner 40, eine Datenverarbeitungseinheit 50, einen Speicher 52, nämlich einen Direktzugriffsspeicher und eine Langzeitspeichervorrichtung 53, nämlich einen Platten-Speichermechanismus enthalten. Die Verarbeitungseinheit 50 hat einen Lese- und Schreibzugriff sowohl zu dem Speicher 52 als auch zu dem Langzeitspeicher 53. Der Speicher 52 kann eine Kopie eines Anwendungsprogrammes zu dessen Ausführung durch die Verarbeitungseinheit 50 aufnehmen. Der Speicher 52 kann ferner eine Nachrichtendatei für das Anwendungsprogramm aufnehmen. Die Verarbeitungseinheit 50 ist durch Kommunikationsverbindungen 44 mit den jeweiligen Benutzerstationen 42 verbunden.

Der Speicher 52 ist, wie in Fig. 4 schematisch dargestellt ist, vorzugsweise derart in logische Abschnitte unterteilt, daß ein Benutzer sowohl exklusiven Lese- und Schreibzugriff zu einem Privatabschnitt 54 des Speichers (hiernach Privatspeicher 54 genannt) hat, als auch Lesezugriff zu einem Gemeinschaftsabschnitt 56, den der Benutzer mit anderen Benutzern teilt (hiernach Gemeinschaftsspeicher 56 genannt) hat. Der Gemeinschaftsspeicher 56 nimmt Anwendungsprogramme und Nachrichtendateien auf, die von mehreren Benutzern verwendet werden können. Der Privatspeicher 54 kann privat angelegte Anwendungsprogramme und Nachrichtendateien aufnehmen. Fig. 4 zeigt drei Privatspeicherabschnitte 54 (gekennzeichnet Benutzer/1, Benut-

zer/2 und Benutzer/k) und den Gemeinschaftsspeicher (gekennzeichnet "alle Benutzer"). Benutzer/1 kann Lese- und Schreibzugriff nur zu dem mit Benutzer/1 korrespondierenden Privatspeicher 54 und Lesezugriff zu dem Gemeinschaftsspeicher 56 haben. In gleicher Weise kann Benutzer/k Lese- und Schreibzugriff nur zu dem zu Benutzer/k korrespondierenden Privatspeicher 54 und Lesezugriff zu dem Gemeinschaftsspeicher 56 haben. Schreibzugriff zu dem Gemeinschaftsspeicher wird nur in besonderen Umständen geschaffen, nämlich für einen Systemverwalter.

Eine geeignete Datenverarbeitungsanlage 38 ist die durch die Firma Prime Computer of National, Massachusetts, vertriebene Prime 50 Computerserie. Zu Darstellungszwecken konzentriert sich die nachfolgende Beschreibung auf eine Verwirklichung unter Verwendung dieses Systems. Solch eine Prime 50 Datenverarbeitungsanlage kann unter Verwendung der Programmiersprache "SPL" (die dem PL/I ähnlich ist) konfiguriert werden um die Datenstrukturen und die weiter unten beschriebenen Prozesse zu implementieren. Die Speicherstruktur hat vorzugsweise eine ringförmige Architektur, aber andere Standardspeicherstrukturen können auch geeignet sein. Die obige Beschreibung in Verbindung mit den Fig. 1-4 hat rein beschreibenden Charakter. Andere, einen Gemeinschafts- und Privatspeicher aufweisende Mehrfachbenutzer-Computersystem sind ebenso zur Ausführung der Erfindung geeignet. Basierend auf dieser technischen Lehre weiß der Durchschnittsfachmann, wie er die hierin beschriebenen Datenstrukturen und Fließdiagramme abändern muß um sie auf andere Architekturen oder andere Systeme einzusetzen zu können.

Um mehrsprachige Dialoge mit auf einem oben beschriebenen System laufenden Anwendungsprogrammen zu haben, hat ein Anwendungsprogramm zumindest einen korrespondierenden Nachrichtenteil. Eine Nachrichtendatei enthält Eingabe- und Ausgabenachrichten für ein Anwendungsprogramm in einer natürlichen Sprache. Nachrichtendateien, die Nachrichten in verschiedenen natürlichen Sprachen bereithalten, werden vorzugsweise separat gehalten. Eine Nachrichtendatei sollte in einer Art und Weise bezeichnet werden, so daß der Name einer Nachrichtendatei von dem Anwendungsprogramm und der Sprache einfach bestimmt werden kann. Ein geeignetes Format ist "<Anwendungsname>. <Sprache>. msg". Der Fachmann kennt auch andere geeignete Formate. Das ausgewählte Format hängt oft von dem Rechner ab, auf dem das System verwirklicht ist.

Eine geeignete Nachrichtendatei, nämlich "Beispiel. englisch. msg" ist in Tabelle I gezeigt und enthält zwei Nachrichten:

Max-Benutzer-überschriften "Das System erlaubt keine Benutzer in diesem Moment".

Unmöglich-zu-add-oder "Ihr ursprüngliches Verzeichnis kann momentan nicht erreicht werden".

TABELLE I

Derartige Nachrichtendateien können von einer Nachrichtenquelldatei entwickelt werden, die die Nachrichten für alle Sprachen enthalten. Die Quelldatei kann compiliert werden um verschiedene Nachrichtendateien zu erzeugen. Eine Quelldatei hilft die Genauigkeit von Übersetzungen einer Nachricht dadurch zu gewährleisten, daß sie diese in einer gemeinsamen Datei plaziert, so daß ein Benutzer sie gleichzeitig betrachten

kann. Eine derartige Quelldatei ist in Tabelle II gezeigt.

Wenn die Nachrichten-Quelldatei kompiliert ist, dann kann eine Überführung von Mnemoniken (wie "Max-Benutzer-überschritten") in Nummern ebenso durchgeführt werden um Speicherplatz zu sparen. Die in der Nachrichten-Quelldatei verwendeten Sprachnamen und die Namen der Nachrichtendateien können andere Kennzeichen des Sprachennamens sein und müssen nicht dem korrekten Namen, zum Beispiel englisch, entsprechen. Diese anderen Kennzeichnungen können Abkürzungskodes, wie den Iso-Standard 639 oder den Länderkode enthalten.

LANGUAGES

- 0 "English"
- 1 "français"
- 2 "Deutsch" max users exceeded
- 0 "System not admitting users at this time"
- 1 "Nombre maximum d'usagers depasse."
- 2 "Maximale Benutzer-Anzahl wird überschritten." unable to at or
- 0 "Unable to attach to your initial directory."
- 1 "Incapable de vous relier a votre boite initiale."
- 2 "Das Aufprüfen auf die Zugangs ist nicht möglich."

Tabelle II

Wenn ein Anwendungsprogramm (oder eine Wörterbuch) in den Speicher geladen ist, dann muß auch eine Nachrichtendatei für das Programm in den Speicher geladen werden. Um das Anwendungsprogramm, die Nachrichtendatei und den Benutzer miteinander zu verbinden, so daß das Anwendungsprogramm in die Lage versetzt wird Nachrichten von einer Nachrichtendatei abzurufen, muß eine Verbindung zwischen dem Anwendungsprogramm und seiner Nachrichtendatei aufgebaut werden. Für einen Benutzer speichert eine Verbindungstabelle 60 (Fig. 5) Verbindungen für von dem Benutzer geladene Programme.

In dem Privatspeicher 54 steht für jeden Benutzer für private Anwendungsprogramme eine Verbindungstabelle 60 (hiernach die private Verbindungstabelle 60 genannt) bereit. Der Gemeinschaftsspeicher 56 (Fig. 3) enthält ebenso eine Verbindungstabelle 60, aber diese Verbindungstabelle wird für Gemeinschafts-Anwendungsprogramme (hiernach Gemeinschafts-Verbindungstabelle 60 genannt) bereitgestellt.

Eine zusätzliche Verbindungstabelle 60 für das Betriebssystem, welches sich von der Gemeinschafts-Verbindungstabelle unterscheidet, kann auch bereitgestellt werden. Aus Darstellungsgründen wird für die nachfolgende Beschreibung der Arbeitsweise des Systems angenommen, daß eine separate Verbindungstabelle für das Betriebssystem bereit gestellt ist. Nichts desto trotz ist diese Beschreibung rein illustrativ und der Durchschnittsfachmann kennt andere, gleichwertige Alternativen.

Eine Verbindung 65 einer Verbindungstabelle 60 enthält eine Kennzeichnung des Anwendungsprogrammes, der Sprache für eine Nachrichtendatei und der Speicherstelle der Nachrichtendatei im Speicher. Eine bevorzugte Form einer Verbindung enthält ein Programmfeld 62, ein Sprachfeld 68 und einen Zeigerfeld 66. Ein Programmfeld 62 identifiziert ein Anwendungsprogramm, welches den Namen des Anwendungsprogrammes oder alternativ den Bereich der Speicherstelle seines Verfahrenskodes im Speicher verwendet. Ein

Sprachfeld 68 kennzeichnet die natürlichen Sprachen, in denen die Nachrichten der geladenen Nachrichtendateien geschrieben sind. Ein Sprachfeld 68 kann den Namen der Sprache, eine Nummer oder alternativ ein Mnemonikkode speichern und hat eine Größe, die von der Implementierung abhängt. Ein Zeigerfeld 66 enthält einen Zeiger zu der Speicherstelle im Speicher, an welcher die Nachrichtendatei geladen ist.

In einer Verbindungstabelle 60 ist eine Verbindung aufgebaut, wenn eine Nachrichtendatei für ein Anwendungsprogramm in einem Gemeinschafts- oder Privatspeicher 54 geladen ist. Die durch eine aufgebaute Verbindung angezeigte Sprache ist "installiert" für das Anwendungsprogramm. Die Installation von Sprachen und an deren Betriebsweisen des Systems wird weiter unten ausführlich beschrieben werden.

Die Fig. 6—8 sind Diagramme von Datenstrukturen, wie sie im Gemeinschaftsspeicher 56 des Datenverarbeitungssystems 38 vorkommen. Diese Strukturen stellen die Grundunterstützung zum Betreiben des Systems im mehrsprachigen Dialog für Gemeinschaftsanwendungen zur Verfügung. Diese Strukturen enthalten eine System-Informationen-Tabelle 70 (Fig. 6), eine System-Sprach-Tabelle 76 (Fig. 7) und eine System-Freigabe-Tabelle 90 (Fig. 8). Die System-Informationen-Tabelle 70 (Fig. 6) hält Informationen bezüglich des Status der mehrsprachigen Dialog-Kapazitäten bereit. Die System-Sprach-Tabelle 76 (Fig. 7) wird zum Abgleich von Sprachnamen mit Sprachkodes verwendet. Die System-Freigabe-Tabelle 90 (Fig. 8) hält Informationen bezüglich des Status der Freigabe des Systems bereit.

Die System-Informationen-Tabelle 70 (Fig. 6) wird nun ausführlich beschrieben. Die System-Informationen-Tabelle 70 enthält ein Feld 72, welches die maximale Anzahl von Sprachen anzeigt, die in dem System installiert werden kann. Die Tabelle 70 enthält ebenso ein installiertes Sprachfeld 74, welches die Anzahl der natürlichen Sprachen anzeigt, die momentan auf dem System installiert sind. Sowohl das Feld 72 als auch das Feld 74 haben die Größe eine Bytes. Der Wert des installierten Sprachfeldes 74 ist durch den Wert des maximalen Sprachfeldes 72 begrenzt. Eine System-Sprach-Tabelle 76, wie sie weiter unten in Verbindung mit Fig. 7 beschrieben wird, kann ebenso in der System-Informationen-Tabelle 70 enthalten sein.

Die System-Informationen-Tabelle 70 enthält weiterhin ein Zeigerfeld 78, welches verwendet wird um die Speicherstelle der Betriebssystem-Verbindungs-Tabelle 60 im Speicher 52 zu kennzeichnen. In Tabelle 70 wird ebenso ein Zeigerfeld 79 geschaffen um die Speicherstellen der Gemeinschafts-Verbindungs-Tabelle 60 (Fig. 5) im Speicher 52 (Fig. 3) zu kennzeichnen. Obwohl es für beide Zeigerfelder 78 und 79 möglich ist, durch die aktuelle Verbindungstabelle 60 zu der sie zeigen versetzt zu werden, ist es aus zwei Gründen vorteilhaft, die Verbindungstabellen 60 von der System-Informationen-Tabelle 70 zu trennen. Diese Trennung erlaubt es der Verbindungstabelle 60 dynamisch zugeordnet zu werden, während sie der System-Informationen-Tabelle 70 erlaubt, statisch zugeordnet zu werden. Weil die System-Informationen-Tabelle 70 eine feste Größe hat, ist für diese Tabelle eine dynamische Zuordnung nicht erforderlich. Durch das Vorhandensein von verschiedenen Verbindungstabellen für das Betriebssystem und für Gemeinschaftsprogramme wird der Zugriff zu der Betriebssystem-Verbindungstabelle 60 durch das System eingeschränkt, so daß eine größere Sicherheit der Echtheit des Betriebssystems geschaffen wird.

Die System-Sprach-Tabelle 76, dargestellt in Fig. 7, hat Einträge 80, die einen Sprachkode 82 und einen korrespondierenden Sprachnamen 84 enthalten. Ein Beispiel für einen Sprachnamen 84 ist "englisch".

Der Sprachkode 82 ist vorzugsweise ein numerischer Kode, aber auch andere Kodes, wie Mnemonikkodes, wie "ENG" oder Zahlen können auch verwendet werden. Das Sprachkodefild 82 ist vorzugsweise ein Byte. Auf Eintragungen in dieser Tabelle 76 kann entweder die Sprachkodeeintragung 82 oder die Sprachname-Eintragung 84 zugreifen. Das Sprachnamenfeld 84 ist aus 10 dynamisch zugeordneten Zeichen gemacht.

Der System-Sprach-Tabelle 76 ist genug Speicherplatz zugeordnet um die maximale Anzahl von Eintragungen 80 zu speichern, die der maximal spezifizierten Anzahl von Sprachfeldern 72 entspricht. Somit hat die System-Informationen-Tabelle 70 eine feste, vorbestimmte Größe. Feld 72 wird während der Initialisierung des Systems auf einen vorbestimmten Wert, zum Beispiel "16" gesetzt.

Die System-Freigabe-Tabelle 90 (Fig. 8) ist geschaffen um den Status der Freigabe der Dialog-Kapazität des Systems anzuzeigen, wie oben bereits beschrieben wurde. Der Status des Dialoges kann entweder "private Freigabe" oder "volle Freigabe" sein. Das System ist privat freigegeben, wenn das System in der Lage ist mit einem Benutzer durch ein in den Privatspeicher 54 geladenes Anwendungsprogramm oder ein Wörterbuch in mehreren natürlichen Sprachen zu kommunizieren. Das System ist voll freigegeben, wenn das System in der Lage ist, mit einem Benutzer über gemeinschaftliche und private Anwendungsprogramme und über das Betriebssystem in mehreren natürlichen Sprachen zu kommunizieren.

Der Inhalt der Tabelle 90 unterscheidet sich in den verschiedenen Freigabezuständen, wie in den Kolonnen 91 bzw. 93 in Fig. 8 gezeigt ist. Feld 92 wird verwendet um anzuzeigen, ob das System zumindest privat freigegeben ist. Das heißt, Feld 92 enthält einen Falschwert, zum Beispiel 0, wenn das System nicht für den privaten Gebrauch freigegeben ist. Feld 92 kann durch einen einzelnen Bit realisiert werden, der zwischen einem logischen 0 und einem logischen 1 geschaltet werden kann. Die Verwendung dieses Feldes 92 ermöglicht mehrsprachige Dialoge mit privaten Anwendungsprogrammen (das heißt, das System kann privat freigegeben sein) ohne den Dialog mit dem Betriebssystem und den gemeinschaftlichen Anwendungen (d. h. wenn das System nicht voll freigegeben ist) freizugeben. Diese Fähigkeit ist besonders in Gebieten wertvoll, wo das Betriebssystem und/oder gemeinschaftliche Anwendungen mit nur einer (oder keiner) Dialog-Sprache-Nachrichtendatei versehen ist, aber wo Benutzer private Anwendungsprogramme mit mehreren Nachrichtendateien haben.

Das System-Standard-Sprachenfeld 94 hat die Größe von einem Byte, ist also auch in der System-Freigabe-Tabelle 90 vorhanden, um die natürliche Standard-Sprache des Betriebssystems zum Dialog mit einem Benutzer anzuzeigen. Auch wenn ein Benutzer ein privates oder ein gemeinschaftliches Anwendungsprogramm benutzt, für das die Benutzer Standardsprache nicht erhältlich ist, dann versucht das System diese Standardsprache zu benutzen. Schließlich ist ein Zeigerfeld 96 bereitgestellt um den Speicherplatz der System-Informationen-Tabelle 70 im Speicher 52 anzuzeigen.

Die Strukturen der Fig. 6—8, die sich im Gemeinschaftsspeicher 56 (Fig. 4) wiederfinden, schaffen eine Basisstruktur für mehrsprachige Dialoge mit dem Be-

triebssystem und gemeinschaftlichen Anwendungsprogrammen. Solche Dialoge sind vorzugsweise in einer Weise bereit gestellt, die einfach freigegeben und gesperrt werden können. Zu diesem Zweck wird vorzugsweise nur die System-Freigabe-Tabelle gemäß Fig. 8 als eine permanente Struktur des Systems gepflegt. Daher brauchen andere Informationen und Strukturen nicht zugeordnet werden, wenn ein mehrsprachiger Dialog nicht voll freigegeben ist.

Bezugnehmend auf die Fig. 9—13 werden nun die auf einer Pro-Benutzer-Basis zugewiesenen Datenstrukturen in einer bevorzugten Ausführungsform der Erfindung beschrieben. Die folgenden Datenstrukturen werden für den Benutzer im Privatspeicher 54, zusammen mit einer Privat-Verbindungs-Tabelle 60 bereitgehalten: Ein dem System-Standard-Sprache-Feld 94 (Fig. 8) ähnlichen Benutzer-Standard-Sprachen-Feld 100 (Fig. 9), eine der System-Sprach-Tabelle 76 ähnliche Benutzer-Sprach-Tabelle 110 (Fig. 11—12); eine der System-Informationen-Tabelle 70 ähnliche Benutzerinformationstabelle 130 (Fig. 13) und ein verschachteltes Sprach-Stapelregister 102 (Fig. 10) zur Speicherung verschachtelter Sprachinformationen.

Das Benutzer-Standardsprachen-Feld 100 (Fig. 9) dient der Speicherung der bevorzugten Dialogsprache eines Benutzers und hat eine Größe von einem Byte. Wenn die Struktur 100 einem Benutzer zugewiesen ist, dann benutzt es den Wert des System-Standard-Sprachen-Feldes 94 (Fig. 8), außer es wird durch den Benutzer ausdrücklich geändert. Durch die Ausführung eines Befehles von der Benutzerstation ist der Benutzer in der Lage, diese Standardsprache zu ändern. Dieser Befehl ist vorzugsweise in der Form des Namens der gewünschten Sprache, wie "französisch" auszuführen. Die Schritte dieses Befehles werden später in Verbindung mit Fig. 21 ausführlich beschrieben werden. Wenn das System für mehrsprachige Dialoge freigegeben ist, dann zeigt Feld 100 eine Sprache durch einen Sprachennamen, einen Sprachencode oder durch eine andere Art von Kennzeichnung (Kolonne 101) an. Ansonsten enthält es einen Nicht-Sprachwort, zum Beispiel — 1 (Kolonne 99). Das Benutzer-Sprachfeld 100 ist vorzugsweise im Speicher 52 abgespeichert, welcher nicht re-initialisiert wird, wenn ein Benutzer-Prozeß re-initialisiert wird. In einer ringförmigen Architektur kann ein Ring diese Bedingung erfüllen. Für das System und gemeinschaftliche Anwendungen ist dieses Feld 100 analog zu Feld 94. Es zeigt eine zu verwendende Standardsprache an, wenn für das Betriebssystem oder Anwendungsprogramme oder Wörterbücher keine andere Sprache spezifiziert oder verfügbar ist. Wenn diese Sprache nicht verfügbar ist, kann das System die Standardsprache benutzen.

Ein verschachteltes Sprach-Stapelregister 102 (Fig. 10) wird einem Benutzer ebenso bereitgestellt. Es kann in Verbindung mit einem Standardmechanismus zur Verschachtelung von Anwendungsprogrammen eingesetzt werden. Dieses verschachtelte Sprach-Stapelregister 102 ist wie ein Stapelregister oder eine ähnliche Struktur wie eine verbundene Liste oder ein Feld aufgebaut. Eine Eingabe 104 in dem verschachtelten Sprach-Stapelregister 102 enthält ein Feld 106, welches die Sprache anzeigt, in der ein zuvor gelaufenes gemeinschaftliches Anwendungsprogramm mit dem Benutzer kommuniziert hat, sowie ein Feld 108, welches die Sprache anzeigt, in der ein zuvor gelaufenes privates Anwendungsprogramm mit dem Benutzer kommuniziert hat. In einer Eingabe 24 zeigt eines dieser Felder eine Spra-

che an und die anderen speichern einen Nicht-Sprachwort. Jedes Feld hat vorzugsweise eine Größe von einem Byte.

Eine Benutzer-Sprach-Tabelle 110 (Fig. 11) wird einem Benutzer zur Speicherung von Informationen über verfügbare oder installierte private Dialog-Sprachen bereitgestellt. Das installierte Sprachenfeld 112 ist geschaffen um die Anzahl der Sprachen anzuzeigen, die installiert sind und hat eine Größe von vorzugsweise einem Byte. Weil einer installierten Sprache ein Kode, vorzugsweise eine Zahl beigelegt ist, kann die Anzahl der installierten Sprachen verwendet werden, um den Kode für eine neuerlich installierte Sprache zu erzeugen.

Es kann vorkommen, daß sich der Kode eines Benutzers für eine private Sprache von dem Kode der Systemsprache für die Sprache, die in der System-Sprach-Tabelle 76 gefunden wurde unterscheidet. Deshalb wird eine Benutzersprachen-Zugehörigkeits-Tabelle 114 geschaffen, die die Zugehörigkeit zwischen einem spezifischen Benutzer-Sprachkode und Sprachnamen mit den systemweiten Sprachkodes anzeigt. Die Struktur der Benutzersprachen-Zugehörigkeitstabelle 114 ist in Fig. 12 ausführlicher dargestellt. In der Tabelle 114 kann ein Eintrag 116 entweder unter Verwendung eines Benutzer-Sprachkodes, wie durch Bezugsziffer 118 gezeigt ist, oder unter Verwendung eines Feldes 120 mit einer Größe von etwa 10 Zeichen, die den zugehörigen Sprachnamen anzeigen und eines Feldes 122, welches den durch das Betriebssystem benutzten Sprachkode anzeigt, indiziert werden, wie es durch die System-Sprach-Tabelle 76 (Fig. 7) aufgebaut ist. Die Felder 118 und 122 können eine Größe von etwa 1 Byte haben. Diese Tabelle 114 wird verwendet, weil der von einem Benutzer benutzte Sprachkode zur Darstellung einer Sprache von dem von dem Rest des Systems benutzten Kode abweichen kann.

Die Benutzer-Informationen-Tabelle 130 (Fig. 13) ist der System-Informationen-Tabelle 70 ähnlich. Diese Tabelle wird verwendet, um die benutzerspezifischen Informationen zusammenzufassen und um Verbindungen zu den systemweiten Informationen bereitzustellen. Diese Tabelle enthält vier Zeigerfelder 132, 142, 144 und 146. In den Kolonnen 131 bzw. 133 sind repräsentative Werte für ein freigegebenes System und ein nichtfreigegebenes System gezeigt. Das Zeigerfeld 132 zeigt die Speicherstelle des durch den Benutzer verschachtelten Sprach-Stapelregisters 102 (Fig. 10) an. Das Zeigerfeld 142 zeigt die Speicherstelle der privaten Verbindungstabelle 60 (Fig. 5) im Privatspeicher 54 an, welche für einen Benutzer eines Privatspeichers 54 bereitgestellt ist. Das Zeigerfeld 144 zeigt die Speicherstelle der System-Informationen-Tabelle 70 (Fig. 6) im Gemeinschaftsspeicher 56 (Fig. 3) an. Schließlich zeigt das Zeigerfeld 146 die Speicherstelle der Benutzer-Sprach-Tabelle 110 (Fig. 11) an. Eines oder mehrere dieser Zeigerfelder können jederzeit durch die Strukturen, auf die sie zeigen, ersetzt werden. Jedoch wenn der mehrsprachige Dialog des Systems nicht freigegeben ist, dann spart das Separieren dieser Strukturen-Speicherplatz. Wenn die Benutzer-Informationen-Tabelle 130 Strukturen mit unterschiedlicher Größe enthält, dann hat es keine feste Größe mehr, was möglich aber nicht bevorzugt ist.

Die verbleibenden Felder der Benutzer-Informationen-Tabelle 130 (Fig. 13) enthalten ein Feld 134, welche die Dialogsprache für das momentan laufende private Anwendungsprogramm und ein korrespondierendes Feld 136 für gemeinschaftliche Anwendungsprogramme an-

zeigt. Ein Feld 138 zeigt die momentane maximale Anzahl von Privatsprachen an, in denen ein Benutzer mit dem System kommunizieren kann. Diese Anzahl kann dynamisch, vorzugsweise in Schritten von 4, expandieren, soweit notwendig. Schließlich wird ein Feld 140 verwendet, um die maximale Menge von Privatspeicher 54 anzuzeigen, die ein Benutzer haben kann. Alle diese Felder haben eine Größe von vorzugsweise 1 Byte.

Der mehrsprachige Betrieb einer bevorzugten Ausführungsform der Erfindung wird nun in Verbindung mit den Fig. 14 bis 33 beschrieben.

Allgemein kann die Betriebsweise in folgender Weise beschrieben werden. Ein Anwendungsprogramm kann Eingabe- und Ausgabe-Nachrichtenbefehle enthalten, die den in einsprachigen Systemen verwendeten Befehlen ähnlich erscheinen. Gleichwohl kann ein Anwendungsprogramm nur die gewünschten Nachrichten anzeigen, die eine Nachrichtenzahl oder ein Mnemonik verwenden. Das Betriebssystem eines Computers oder eines mehrsprachigen Prozessors empfängt diesen Befehl und bestimmt die Nachrichtendatei, von der die angezeigte Nachricht abgerufen werden soll. Die oben bevorzugten Datenstrukturen und die hiernach bevorzugten Prozessoren unterstützen diesen Dialog durch ein Erkennlichmachen des mehrsprachigen Dialoges für den Anwendungsprogrammierer und Benutzer.

Die Konfiguration eines Systems für voll freigegebene, mehrsprachige Dialoge mit Anwendern, wie es beispielhaft durch die Fig. 1 bis 4 gezeigt ist, wird in Verbindung mit dem Fließdiagramm gemäß Fig. 14 ausführlicher beschrieben werden. Der Konfigurationsprozeß enthält die Initialisierung von Systemvariablen oder Argumenten und das Laden von Nachrichtendateien für das Betriebssystem, sowie den Aufbau von Verbindungen hierzu. Diese Konfiguration beginnt mit der Suche (Schritt 150) von geeigneten Nachrichtendateien für das Betriebssystem. Eine Zählvariable oder ein Argument, z. B. Name "X", wird als die Anzahl der gefundenen Sprachen gesetzt. Falls erwünscht, kann die Zahl "X" durch einen Benutzer einfach gesetzt werden, um die Suche in Schritt 150 zu erleichtern. Diese Option ist wünschenswert, wenn die Anzahl der Sprachen bekannt ist. Dann wird das System für den Gebrauch mit der Anzahl "X" Sprachen in Schritt 152 konfiguriert. Der Schritt 152 der Konfiguration des Systems zum Gebrauch von "X" Sprachen wird ausführlicher in Verbindung mit Fig. 15 beschrieben. Wenn der Schritt 152 der Konfiguration des Systems nicht erfolgreich ist, wie Schritt 154 herausfindet, dann wird ein Fehler signalisiert (Schritt 158). Wenn die Konfiguration des Systems erfolgreich ist, dann wird die Durchführung mit Schritt 156 fortgesetzt, indem Nachrichtendateien für die verfügbaren Sprachen geladen werden. Der Prozeß der Ladung einer Nachrichtendatei wird ausführlicher in Verbindung mit Fig. 17 beschrieben.

Der Prozeß der Konfiguration des Systems für mehrsprachige Dialoge mit einer Anzahl "X" von natürlichen Sprachen wird durch das Fließdiagramm gemäß Fig. 15 dargestellt. In diesem Prozeß wird in Schritt 160 zuerst bestimmt, ob das System bereits konfiguriert oder vorzugsweise durch Überprüfung des Zeigerfeldes 96 (Fig. 8), welches anzeigt, ob die System-Informationen-Tabelle 70 initialisiert wurde, voll freigegeben wurde. Weil ein konfiguriertes System nicht rekonfiguriert werden sollte solange andere Benutzer es verwenden, wird ein Fehler signalisiert (Schritt 162), wenn eine Rekonfiguration versucht wurde. Anderenfalls wird das System in Schritt 164 durch Verminderung des Speicherplatzes

für die System-Informationen-Tabelle 70 (Fig. 6) sowie die Verbindungstabellen 60 (Fig. 5) für das Betriebssystem und für gemeinschaftliche Anwendungsprogramme konfiguriert.

Die Größe der System-Sprach-Tabelle 76 (Fig. 7), für die eine Verminderung (Schritt 164) durchgeführt wurde, wird durch die Anzahl der durch den Wert "X" angezeigten Sprachen bestimmt. Durch diesen Verminderungs-Schritt 164 werden die Zeigerfelder 78 und 79 der System-Informationen-Tabelle 70 (Fig. 6) gesetzt. Das Zeigerfeld 96 (Fig. 8) der System-Freigabe-Tabelle 90 wird ebenso in Schritt 164 gesetzt. Die verbleibenden Werte der system-Informationen-Tabelle 70, d. h. das installierte Sprachenfeld 74 und das maximale-Anzahl-der-Sprachen-Feld 72 werden in Schritt 166 jeweils zu der Anzahl der installierten Sprachen und zu der maximalen Anzahl der Sprachen, z. B. des Wertes von "X" gesetzt. Anschließend wird der mehrsprachige Dialog für private Anwendungsprogramme in Schritt 168 durch Besetzen des Feldes 92 der System-Freigabe-Tabelle 90 mit einem "Wahr"-Wert (Schritt 168) freigegeben. Nach der Beendigung des Konfigurationsschrittes bleiben die Verbindungstabellen 60 leer und es gibt keine Sprachworte in den Sprachtabellen. Weiterhin müssen Nachrichtendateien noch installiert werden (Feld 72) und eine Standard-Dialogsprache (Feld 94) muß noch gesetzt werden.

Die Standard-Dialogsprache des Betriebssystems (Feld 94) kann durch eine Person mit ausreichender Systemauthorisation, wie einem Systemverwalter, durch Verwendung eines in Fig. 16 beschriebenen Prozesses geändert werden. Der Systemverwalter ruft den Prozeß auf, der eine gewünschte, neue Dialogsprache anzeigt. Weil die Standardsprache des Systems nicht geändert werden sollte, außer das System wird konfiguriert, sollte solch eine Konfiguration geprüft werden (Schritt 170 ähnlich dem Schritt 160 der Fig. 15) und ein Fehler angezeigt werden (Schritt 172), wenn das System nicht konfiguriert ist. Wenn das System konfiguriert ist, dann ist bestimmt (Schritt 174) ob die Nachrichtendatei für die gewünschte Dialogsprache in den Speicher geladen ist. Wenn die gewünschte Dialogsprache nicht in den Speicher geladen ist, dann wird ein Fehler signalisiert (Schritt 176) und der Betrieb wird abgebrochen. Anderenfalls wird das Feld 94 der standard-Dialogsprache des Betriebssystems (Fig. 8) gesetzt (Schritt 178) um die gewünschte Sprache anzuzeigen. Diese Standardsprache des Systems ist nicht die gleiche wie die Standard-sprache des Benutzers. Erstere kann nur durch den Systemverwalter oder dergleichen verändert werden; letztere durch den Benutzer, obwohl beide dieselbe Sprache anzeigen können.

Wenn das System konfiguriert wurde, können Anwendungsprogramme geladen und ausgeführt werden und Verbindungen werden zwischen Programmen und Nachrichtendateien hergestellt.

Der Prozeß zur Ladung einer Nachrichtendatei für eine Dialogsprache für eine gemeinschaftliche Anwendung wird nun in Verbindung mit Fig. 17 beschrieben. Für diesen Prozeß wurde die gewünschte Dialogsprache für das Anwendungsprogramm bereits aufgebaut. Der erste Schritt dieses Prozesses ist Schritt 180 (ähnlich dem Schritt 160), der bestimmt, ob das System für einen mehrsprachigen Dialog freigegeben ist. Wenn das System nicht voll freigegeben ist, wird ein Fehler signalisiert (Schritt 182) und das Verfahren wird abgebrochen. Wenn in Schritt 180 festgestellt wird, daß das System bereits für einem mehrsprachigen Dialog konfiguriert

wurde, dann wird der Sprachkode für die gewünschte Dialogsprache in der System-Sprach-Tabelle 76 (Fig. 7) in Schritt 184 gesucht. Wenn der Kode nicht gefunden wird und wenn die Tabelle 76 voll ist, was in Schritt 186 festgestellt wird, dann wird ein Fehler signalisiert (Schritt 188), der anzeigt, daß auf dem System bereits die maximale Anzahl von Sprachen installiert wurde.

Wenn der Sprachkode in Schritt 184 nicht gefunden wird, aber die System-Sprach-Tabelle 76 nicht voll ist, dann wird ein Eintrag 80 zu der System-Sprach-Tabelle 76 (Schritt 190) für diesen Sprachkode und seinen zugehörigen Sprachnamen hinzugefügt. Nachdem der Sprachkode für die Nachrichtendatei der nachgesuchten Dialogsprache in der System-Sprach-Tabelle gefunden wurde (Schritt 184) oder in ihr in Schritt 190 plazierte wurde, wird der Speicherplatz im Gemeinschaftsspeicher 56 für diese Nachrichtendatei vermindert (Schritt 192). Im Anschluß an den Schritt 192 der Verminderung des Speicherplatzes wird die Gemeinschafts-Verbindungstabelle 60 (Fig. 5) gesucht, um zu bestimmen, ob ein alter Zeiger zu einer Nachrichtendatei für diese Dialogsprache existiert. Wenn ein alter Zeiger gefunden wird, wird dieser vorübergehend durch Schritt 194 gesichert. Im Anschluß an die Sicherung des alten Zeigers in Schritt 194 wird der neue Zeiger zu dieser Nachrichtendatei in der Verbindungstabelle 60 (Schritt 196) gespeichert, der wiederum von einem Schritt 198 gefolgt wird, der einen zuvor in Schritt 194 gespeicherten, alten Nicht-Nullzeiger rück-zuordnet.

Nachrichtendateien für ein gemeinsames Anwendungsprogramm sind normalerweise nicht geladen, außer das Anwendungsprogramm ist in einem Gemeinschaftsspeicher 56 geladen. Der oben beschriebene Prozeß nimmt an, daß ein gemeinschaftliches Anwendungsprogramm in den Speicher 52 geladen ist und daß die Nachrichtendatei für die gewünschte Dialogsprache existiert. Die Existenz einer gemeinschaftlichen Nachrichtendatei wird in Schritt 150 (Fig. 14) für Nachrichtendateien des Betriebssystems überprüft und Schritt 206 (Fig. 18) wird für gemeinschaftliche Anwendungsprogramme später beschrieben werden.

Ein gemeinschaftliches Anwendungsprogramm, nicht jedoch das Betriebssystem wird normalerweise durch eine Person mit ausreichender Authorisation, wie ein Systemverwalter, durch die Einführung eines Ladebefehles geladen. Der Prozeß der Ladung ist durch das Fließdiagramm von Fig. 18 skizziert. Das Laden eines gemeinschaftlichen Anwendungsprogrammes als Antwort auf einen Ladebefehl beginnt mit Schritt 204, der das Anwendungsprogramm in den Gemeinschaftsspeicher 56 lädt. Als nächstes wird in Schritt 205 überprüft, ob ein mehrsprachiger Dialog mit dem System durch Überprüfung des Zeigerfeldes 96 der System-Freigabetabelle 90 (Fig. 8) voll freigegeben ist. Wenn ein mehrsprachiger Dialog nicht voll freigegeben ist, dann ist die Ladeoperation abgeschlossen. Anderenfalls wird die Operation mit Schritt 206 fortgeführt, der die verfügbaren Nachrichtendateien für das gemeinschaftliche Anwendungsprogramm bestimmt. Jede verfügbare Nachrichtendatei wird dann in dem Gemeinschaftsspeicher 56 (Schritt 208) gemäß des oben in Verbindung mit der Fig. 17 beschriebenen Prozesses geladen.

Wenn die Prozesse gemäß der Fig. 14 und 15 erfolgreich abgeschlossen wurden, dann ist das System für einen mehrsprachigen Dialog mit dem Betriebssystem und jedem anderen durch den Prozeß gemäß Fig. 18 geladenen, gemeinschaftlichen Anwendungsprogramm voll freigegeben. Die Freigabe eines mehrsprachigen

Dialoges mit privaten Anwendungsprogrammen wird in Verbindung mit den Fig. 19 und 20 noch beschrieben werden.

Um einen mehrsprachigen Dialog für private Anwendungsprogramme freizugeben, wird Schritt 210 (Fig. 19) der Initialisierung des Freigabefeldes 92 (Fig. 8) durchgeführt, um das Feld 92 zu setzen und um anzuzeigen, daß ein Mehrfachdialog für private Anwendungsprogramme freigegeben ist. Dieser Schritt 210 der Initialisierung ist derselbe wie Schritt 168, der als ein Teil der Systemkonfiguration durchgeführt wird (Fig. 15). Der Initialisierungsschritt 210 kann auch durchgeführt werden, um einen mehrsprachigen Dialog mit privaten Anwendungsprogrammen freizugeben, wenn das System für solche Dialoge mit dem Betriebssystem und anderen gemeinschaftlichen Anwendungsprogrammen nicht konfiguriert ist, obwohl dadurch das System privat, aber nicht voll freigegeben wird. Damit ein derartiger Freigabeprozess durchgeführt werden kann, sollte ein Benutzer eine geeignete Berechtigung zum Zugriff auf das Freigabefeld 92 haben, nämlich die Berechtigung eines Systemverwalters oder dergleichen. Dieser Prozeß wird durchgeführt, bevor andere Benutzer mit privaten Anwendungsprogrammen in mehreren Sprachen kommunizieren.

Wenn ein Benutzer beginnt, normalerweise durch "Einloggen" in das System von einer Benutzerstation aus, das Datenverarbeitungssystem zu benutzen, dann werden die Pro-Benutzer Datenstrukturen, insbesondere die Benutzer-Informations-Tabelle 110 (Fig. 11), durch den durch das Fließdiagramm gemäß der Fig. 20 beschriebenen Prozeß initialisiert. Die Initialisierung beginnt mit dem Schritt 220, der die Sprachcodefelder 134 und 136 und den Zeiger 144 zu der System-Informationen-Tabelle 70 (Fig. 6) der Benutzer-Informationstabelle 130 (Fig. 13) initialisiert. Die Sprachcodes der Felder 134 und 136 werden mit einem Nicht-Sprachwort initialisiert und der Zeiger 144 der System-Informationen-Tabelle wird auf die Speicherstelle der system-Informationen-Tabelle 70 gesetzt, die aus dem korrespondierenden Feld 96 der System-Freigabe-Tabelle 90 (Fig. 8) erhalten wird. Wenn das System nicht zumindest für einen mehrsprachigen Dialog privat freigegeben ist (er kann ebenso voll freigegeben sein), was in Schritt 222 festgestellt wird, dann werden die verbleibenden Pro-Benutzerfelder mit nicht freigegebenen standardwerten besetzt. D.h. die maximale Anzahl von Benutzersprachen 138, die maximale Menge von Privatspeicher 140, der Zeiger 142, der zu der privaten Verbindungstabelle 60 zeigt und der Zeiger 146, der zu der Benutzersprache 110 zeigt, werden alle zu beispielsweise "0", "0" "Null" bzw. "Null" (Schritt 224) gesetzt. In dieser Weise wird Speicherplatz weder für die private Verbindungstabelle 60 noch für die Benutzer-Sprachtable 110 zugeordnet und somit wird Speicherplatz eingespart. Wenn ein Mehrfachdialog für private Anwendungsprogramme freigegeben wurde, wie in Schritt 222 festgestellt wird, dann wird der Anwender-Sprachtable 110 und der privaten Verbindungstabelle 60 Speicherplatz zugewiesen (Schritt 226). Die Zeiger 142 und 146, die zu den Speicherstellen im Speicher zeigen, die jeweils den beiden Strukturen zugeordnet wurden, werden gesetzt. In Schritt 228 wird das Feld 138, welches die maximale Anzahl von Benutzersprachen anzeigt, mit einem vorbestimmten festen Wert, z. B. 4, besetzt. Die maximale Menge von privatem Speicherfeld 140 wird in ähnlicher Weise initialisiert. Das Verfahren fährt nun mit Schritt 230 fort, der einen Eintrag 116 in der Benutzersprache-

Korrespondenztabelle 114 für den Sprachcode 122 und den Sprachnamen 120 der Standard-Dialogsprache des Betriebssystems durchführt, den er in Feld 94 (Fig. 8) der System-Informationen-Tabelle 90 findet. Feld 118 wird gesetzt, um den ersten Sprachcode, beispielsweise "1" zu halten. Feld 112 der Benutzer-Sprachtable 110 (Fig. 11) wird mit "0" besetzt. Die Initialisierung der Pro-Bediener-Datenstrukturen wird durch Schritt 234 beendet, der die Standard-Dialogsprache des Benutzers des Betriebssystems setzt und das Benutzersprachfeld 100 (Fig. 9) initialisiert.

Nachdem ein Benutzer initialisiert worden ist, ist das Datenverarbeitungssystem voll freigegeben (falls die Systemkonfiguration abgeschlossen ist) oder anderenfalls privat freigegeben für einen mehrsprachigen Dialog mit dem Benutzer. Wenn das System überwiegend für private Benutzer freigegeben ist und nicht für Gemeinschaftsanwendungen konfiguriert ist, dann ist auch der gemeinschaftlichen Datenstruktur kein Speicherplatz zugewiesen, außer für die System-Freigabe-Tabelle 90 (Fig. 8). In dieser Tabelle zeigt das private Freigabefeld 92 an, daß ein mehrsprachiger Gebrauch für private Anwendungen freigegeben ist. Die Standard-Dialogsprache für Betriebssysteme (Feld 94) wird mit einem Nicht-Sprachwort, z. B. "1" besetzt, der die Speicherstelle der System-Informationen-Tabelle 70 anzeigende Zeiger 96 wird mit einem "Null" -Wort besetzt. Die Worte in der Benutzer-Informationen-Tabelle 130 werden mit den in Spalte 133 (Fig. 13) angezeigten Worten und wie bereits oben in Verbindung mit Fig. 20 beschrieben, initialisiert. Die private Systeminformation für einen Benutzer (Fig. 9 bis 13) ist abhängig von der Konfiguration der gemeinschaftlichen Systeminformation (Fig. 5 bis 8) zur Zeit der Initialisierung für diesen Benutzer.

Eine weitere Fähigkeit, die dieses Datenverarbeitungssystem vorzugsweise aufweist, ist daß ein Benutzer die Möglichkeit hat, die im Benutzersprachfeld 100 (Fig. 9) gefundene Standardsprache des Benutzers (vgl. Fig. 21) zu wechseln. Der Benutzer kommuniziert mit dem Betriebssystem und mit Anwendungsprogrammen, die ohne die Spezifikation einer alternativen Sprache aufgerufen wurden, in der Benutzer-Standardsprache. Wenn der Benutzer einen Wechsel der Benutzer-Standardsprache anordnet, wird zuerst festgestellt (Schritt 370), ob die Nachrichtendatei für die gewünschte Dialogsprache des Betriebssystems in dem Speicher überführt worden ist. Wenn es nicht in dem Speicher überführt worden ist, dann wird in Schritt 372 ein Fehler signalisiert. Anderenfalls wird das Benutzer-Standard-Sprachfeld 100 mit der neuen durch den Benutzer angezeigten Sprache besetzt. Diese Sprache wird dann zu der Benutzersprachentabelle 110 in Schritt 376 hinzugefügt. Wenn die Benutzersprachentabelle 110 voll wird, dann wird sie in Schritt 378 in einer Weise erweitert, die hiernach in Verbindung mit Fig. 25 beschrieben werden wird. Dann wird eine Nachricht an den Benutzer über die Benutzerstation 42 ausgesendet, um den Benutzer zu informieren, ob die Standardsprache gesetzt wurde.

Während dem Dialog mit dem Datenverarbeitungssystem kann ein Benutzer private und/oder gemeinschaftliche Anwendungsprogramme aufrufen, die möglicherweise eine gewünschte Dialogsprache anzeigen, die sich von der Benutzer-Standardsprache unterscheidet. Dies ist nützlich, wenn ein Anwendungsprogramm in einer Sprache verfügbar ist, die für das Betriebssystem nicht verfügbar ist, d. h. die keine Benutzer-Standard-Sprache sein kann. Der Prozeß eines Aufrufs einer Anwendung

wird nun in Verbindung mit dem Fließdiagramm gemäß Fig. 22 näher beschrieben.

Wenn ein Anwendungsprogramm aufgerufen ist, wird in Schritt 242 die Bereitschaft des aufgerufenen Anwendungsprogramms geprüft, ausgeführt zu werden. Wenn das System nicht für private, mehrsprachige Dialoge mit privaten Anwendungsprogrammen freigegeben ist, was Schritt 244 überprüft wird und wenn bei dem Aufruf des Anwendungsprogramms eine gewünschte Dialogsprache spezifiziert wurde, was in Schritt 246 festgestellt wird, dann wird in Schritt 248 ein Fehler signalisiert und der Aufruf wird abgebrochen. Wenn keine gewünschte Dialogsprache angezeigt ist, dann wird das Anwendungsprogramm einfach ausgeführt.

Wenn das mehrsprachige System für einen mehrsprachigen Dialog mit privaten Anwendungsprogrammen freigegeben ist, was Schritt 244 feststellt, dann werden die Werte der Sprachfelder 134 und 136 (Fig. 13) auf das durch den Benutzer verschachtelte Sprach-Stapelregister 102 (Fig. 10) geschoben und die Felder 134 und 136 werden mit Nicht-Sprachworten reinitialisiert. Und dann wird in Schritt 250 festgestellt, ob eine gewünschte Dialogsprache angezeigt ist und in Schritt 252 wird ein Argument zu der gewünschten Sprache gesetzt. Wenn keine gewünschte Sprache angezeigt ist, dann wird dieses Argument in Schritt 254 mit einem "Null"-Wort besetzt. Angenommen, entweder in Schritt 252 oder in Schritt 254 ist ein Argument gesetzt worden, noch ist eine Sprache ausgewählt worden und noch wurde für die ausgewählte Sprache eine Nachrichtendatei für das Anwendungsprogramm angezeigt, was die Schritte 257 und 258 feststellen, dann wird in Schritt 260 ein Fehler signalisiert und der Aufruf des Anwendungsprogrammes wird abgebrochen. Wenn keine gewünschte Dialogsprache angezeigt ist und bei Schritt 256 keine Nachrichtendatei gefunden wurde, dann zeigt der zurückgeschickte Sprachkode ein Nicht-Sprachwort an und das geeignete Feld 134 und 136 (Fig. 13) wird besetzt (Schritt 262). Wenn in Schritt 256 eine Nachrichtendatei für eine Dialogsprache gefunden worden ist, dann wird das Sprachfeld 134 und 136 mit der zurückkehrenden Sprache besetzt. Wenn das aufgerufene Anwendungsprogramm ein privates Anwendungsprogramm ist, dann wird das private Sprachfeld 134 mit dem Sprachkode der geladenen Nachrichtendatei besetzt und das gemeinschaftliche Sprachfeld 136 wird mit einem Nicht-Sprachwort besetzt. Die Felder 134 und 136 werden umgekehrt für ein gemeinschaftliches Anwendungsprogramm gesetzt.

Die Fig. 23a und 23b skizzieren den Prozeß der Auswahl einer Sprache und des Findens und des Ladens einer Nachrichtendatei für die ausgewählte Sprache des aufgerufenen Anwendungsprogrammes. Zunächst wird in Schritt 264 bestimmt, ob das Anwendungsprogramm ein gemeinschaftliches Anwendungsprogramm ist, beispielsweise durch Überprüfen der Speicheradresse, in welcher der Prozeßkode geladen ist. Wenn es sich nicht um ein gemeinschaftliches Anwendungsprogramm handelt, dann handelt es sich um ein privates Anwendungsprogramm, für welches das Verfahren der Auswahl einer Sprache und des Findens und Ladens der Nachrichtendatei in Fig. 23b beschrieben ist. Wenn das Anwendungsprogramm ein gemeinschaftliches Anwendungsprogramm ist, dann wird angenommen, daß seine Nachrichtendateien geladen sind, weil nur dann eine Sprache ausgewählt werden braucht.

Für ein gemeinschaftliches Anwendungsprogramm fährt der Prozeß fort mit der Überprüfung des Argu-

mentes, welches die gewünschte Dialogsprache anzeigt (Schritt 266). Wenn eine gewünschte Dialogsprache angezeigt war, dann wird die System-Sprach-Tabelle 76 (Fig. 7) nach dieser Sprache abgesucht (Schritt 268).

Wenn der Suchschritt 268 die gewünschte Sprache nicht findet, dann wird in Schritt 270 ein Nicht-Sprachwort zurückgegeben der anzeigt, daß die gewünschte Dialogsprache für dieses gemeinschaftliche Anwendungsprogramm nicht verfügbar ist, weil, wenn ein gemeinschaftliches Anwendungsprogramm geladen ist, dann sind seine zugehörigen Nachrichtendateien im Gemeinschaftsspeicher 56 geladen und Einträge für die zugehörigen Sprachen sind in der System-Sprach-Tabelle 76 gemacht. Wenn eine gewünschte Dialogsprache für ein gemeinschaftliches Anwendungsprogramm in der System-Sprach-Tabelle 76 nicht gefunden worden ist, dann ist somit diese Sprache für dieses Anwendungsprogramm nicht verfügbar.

Wenn die Sprache in der System-Sprach-Tabelle 76 gefunden worden ist, dann wird die gemeinschaftliche Sprachtabelle 60 überprüft (Schritt 272), um festzustellen, ob die Nachrichtendatei für die gewünschte Dialogsprache in den Gemeinschaftsspeicher 56 des gemeinschaftlichen Anwendungsprogrammes geladen ist. Wenn eine Verbindung für die Sprache und das Programm in der gemeinschaftlichen Verbindungstabelle 60 nicht gefunden worden ist, dann wird ein Nicht-Sprachwort zurückgeschickt (Schritt 274). Wenn in der Verbindungstabelle 60 während des Schrittes 272 eine Verbindung gefunden worden ist, dann wird für diese Sprache ein Eintrag 116 in der Benutzer-Sprach-Tabelle 110 (Fig. 11) plaziert, falls noch keiner vorhanden sein sollte. Wenn die Benutzer-Sprach-Tabelle 110 voll ist, wird sie erweitert, wie weiter unten in Verbindung mit Fig. 25 (Schritt 276) ausführlicher beschrieben werden wird. Wenn, bezugnehmend auf Schritt 266, das Argument der Dialogsprache ein Nicht-Sprachwort war, dann wird die momentane Dialogsprache in Schritt 278 überprüft, um festzustellen, ob eine Dialogsprache momentan im Gebrauch ist, wie durch die Felder 134 und 136 (Fig. 13) angezeigt ist. Wenn eine Dialogsprache momentan in Gebrauch ist, dann wird das Verfahren mit Schritt 280 fortgesetzt, welcher den momentanen Sprachwort mit einem Argument besetzt. Wenn die momentane Sprache für ein privates Anwendungsprogramm vorgesehen ist, dann wird der Sprachkode des Systems von der Benutzersprache-Korrespondenz-Tabelle 114 aufgerufen. Wenn die Sprache für ein gemeinschaftliches Anwendungsprogramm vorgesehen ist, dann wird dieser Wert mit dem Sprachargument besetzt.

Das Verfahren wird mit Schritt 282 fortgesetzt, welcher die Nachrichtendatei in der gemeinschaftlichen Verbindungstabelle 60 sucht, ähnlich dem zuvor beschriebenen Schritt 272. Wenn die Nachrichtendatei in der Verbindungstabelle 60 gefunden worden ist, wird ihr zugehöriger Sprachkode zurückgeschickt (Schritt 284). Anderenfalls wird das Verfahren mit Schritt 286 fortgesetzt. Schritt 286 wird auch durchgeführt, wenn in Schritt 278 festgestellt wird, daß momentan keine Dialogsprache verfügbar ist. Schritt 286 beinhaltet die Überprüfung der Benutzer-Standardsprache, wie sie durch das Benutzer-Standardsprachen-Feld 100 (Fig. 9) angezeigt ist. Wenn der Benutzer-Standardsprachen-Kode ein Nicht-Sprachwort ist, dann wird in Schritt 274 ein Nicht-Sprachwort zurückgeschickt. Anderenfalls wird die gemeinschaftliche Verbindungstabelle 60 nach der Nachrichtendatei dieser Dialogsprache abgesucht,

wie dies zuvor in Schritt 272 getan wurde. Der Prozeß zum Laden der Nachrichtendatei wird dann durch Schritt 274 oder Schritt 276 abgeschlossen, wie dies bereits oben beschrieben ist.

Wenn, wie bereits in Bezug auf Schritt 274 beschrieben wurde, das Anwendungsprogramm als ein privates Anwendungsprogramm bestimmt wurde, dann wird der Prozeß mit Schritt 290 (Fig. 23b) fortgesetzt, indem festgestellt wird, ob eine gewünschte Dialogsprache spezifiziert wurde. Wenn eine Dialogsprache spezifiziert wurde, dann wird die Benutzer-Sprachtable 110 (Fig. 11) in Schritt 292 nach dieser Sprache abgesucht. Wenn diese Sprache in der Benutzer-Sprachtable 110 gefunden worden ist, dann wird in Schritt 294 die private Verbindungstabelle 60 überprüft, um festzustellen, ob die Nachrichtendatei dieser Sprache in den Privatspeicher 54 überführt worden ist. Wenn die Nachrichtendatei überführt worden ist, dann wird in Schritt 296 der Sprachcode zurückgeschickt.

Wenn die Nachrichtendatei für die alternative Sprache nicht überführt worden ist, was in Schritt 294 festgestellt worden ist, dann wird in Schritt 298 festgestellt, ob zuvor bereits nach dieser Nachrichtendatei gesucht worden ist. Wenn zuvor noch nicht nach ihr gesucht worden ist, dann wird in Schritt 300 ein Versuch unternommen, diese Nachrichtendatei dieser Dialogsprache zu überführen. Dieser Versuch 300 wird auch gemacht, wenn die gewünschte Dialogsprache in Schritt 292 nicht in der Benutzer-Sprachtable 110 gefunden worden ist. Wenn der Versuch in Schritt 300, diese Nachrichtendatei zu überführen, erfolgreich ist, dann wird der Sprachcode für diese Nachrichtendatei in Schritt 302 zurückgeschickt. Wenn diese Nachrichtendatei nicht gefunden worden ist, dann wird das Verfahren mit Schritt 304 (wie später beschrieben werden wird) fortgesetzt. Wenn in Schritt 298 ebenso festgestellt wird, daß eine Nachrichtendatei einer Dialogsprache zuvor nachgesucht wurde, dann wird der Prozeß ebenso mit dem noch zu beschreibenden Schritt 304 fortgesetzt.

Wenn, bezugnehmend auf Schritt 290, keine gewünschte Dialogsprache angezeigt ist, dann werden in Schritt 306 die Sprachfelder 134 und 136 überprüft, um festzustellen, ob momentan eine Dialogsprache verwendet wird. Dieser Feststellungsschritt 306 ist dem zuvor in Verbindung mit dem Fließdiagramm gemäß Fig. 23a beschriebenen Schritt 278 ähnlich. Wenn eine Dialogsprache momentan benutzt wird, wird diese ausgewählt und das Verfahren wird mit dem oben bereits beschriebenen Schritt 294 fortgesetzt. Anderenfalls wird das Verfahren mit dem Schritt 304 fortgesetzt, der feststellt, ob durch den Benutzer zum Gebrauch mit dem Betriebssystem bereits eine Dialogsprache ausgewählt worden ist. Dies geschieht durch Überprüfung des Benutzer-Sprachfeldes 100 (Fig. 9). Wenn das Benutzer-Sprachfeld 100 einen Nicht-Sprachwort anzeigt, z. B. "—1", dann wird in Schritt 307 ein Nicht-Sprachcode-Wert zurückgeschickt. Anderenfalls wird die private Verbindungstabelle 60 überprüft, um festzustellen, ob die Nachrichtendatei der durch das Feld 100 angezeigten Sprache in den Privatspeicher 54 überführt worden ist (Schritt 308). Wenn die Datei in den Privatspeicher überführt worden ist, dann wird in Schritt 310 der Sprachcode zurückgeschickt. Anderenfalls wird ein Versuch 312 gemacht, um diese Sprache in den Privatspeicher 54 zu überführen. Wenn der Versuch 312 erfolgreich ist, dann wird in Schritt 310 der Sprachcode der überführten Nachrichtendatei zurückgeschickt. Anderenfalls wird ein Fehlschlag angezeigt und ein Nicht-

Sprachwort, z. B. "—1", wird zurückgeschickt.

Der Prozeß der Überführung einer Nachrichtendatei in einen Pro-Bediener-Privatspeicher 54 (Schritte 300 und 308 gemäß Fig. 22b) wird nun in Verbindung mit Fig. 24 ausführlich beschrieben. Dieser Prozeß hängt vom Empfang eines die gewünschte Sprache anzeigenden Argumentes ab, wobei das Argument in Schritt 320 überprüft wird, um festzustellen, ob das Argument ein Sprachcode oder ein Sprachname ist. Wenn das Argument kein Sprachname ist, dann wird das Verfahren mit Schritt 322 fortgeführt, in welchem der Name der zum Sprachcode zugehörigen Sprache erhalten wird. In Schritt 322 wird unter Verwendung der Benutzersprachen-Korrespondenz-Tabelle 114 (Fig. 12) die Namensfindung durchgeführt. Wenn der Sprachname in Schritt 322 nicht gefunden wird, dann wird eine Fehlermeldung zurückgeschickt (Schritt 324). Anderenfalls wird der Prozeß mit dem Schritt 324 fortgesetzt, indem dem Sprachennamen ein Argument, z. B. "Name" beigesetzt wird, welches entweder gegeben ist oder welches von einem gegebenen Sprachcode erhalten wird. Aus dem Argument "Name", wird ein Dateiname für die gewünschte Nachrichtendatei zusammengesetzt. Das Format dieses Dateinamens hängt von dem speziellen Betriebssystem eines gegebenen Datenverarbeitungssystems ab. Jedoch zeigt dieser Dateiname normalerweise sowohl den Namen des Anwendungsprogrammes, als auch den gewählten Sprachennamen an (Schritt 326). Der Zusammensetzung des Dateinamens folgt der Schritt 328, indem im Langzeitspeicher (z. B. einer Festplatte) eines Datenverarbeitungssystems nach der Nachrichtendatei gesucht wird. Wenn die Nachrichtendatei nicht gefunden wird, dann wird ein Eintrag 64 in die private Verbindungstabelle 60 gemacht (Schritt 330). Dieser Eintragungsschritt 330 enthält das Setzen des Feldes 62, um den Aufruf des Anwendungsprogrammes anzuzeigen, das Setzen des Sprachfeldes 68 auf die gewünschte Dialogsprache und das Setzen des Nachrichtendatei-Zeigerfeldes 66 auf ein Nicht-Nullwort und ein Nicht-Zeigerwort, der anzeigt, daß zwar nach der Nachrichtendatei gesucht wurde, diese jedoch nicht gefunden wurde (Schritt 330). Anschließend wird in Schritt 332 ein Fehler signalisiert. Wenn die Nachrichtendatei nicht gefunden worden ist, dann wird die Benutzersprachen-Korrespondenz-Tabelle 114 (Fig. 12) nach dem Namen der gewünschten Dialogsprache zugehörigen Sprachcode abgesucht (Schritt 334). Wenn in Schritt 334 für diese Eintragung kein Sprachcode gefunden worden ist, dann wird ein Eintrag 116 gemacht, der das Benutzersprachcode-Feld 118 mit dem gegebenen Kode besetzt, der das Sprachen-Namen-Feld 120 mit dem Dialogsprachen-Namen besetzt und der das Feld 122 mit dem Systemcode aus der Systemsprachen-Tabelle 76 (Fig. 7) besetzt, der der Sprache entspricht (Schritt 336). Der Eintragungsschritt 336 enthält auch den Schritt der Erweiterung der Benutzersprachen-Tabelle 110, falls diese voll ist. Diese Tabellen-Erweiterung wird hiernach in Verbindung mit Fig. 25 ausführlich beschrieben werden. Wenn einmal festgestellt ist, daß die Sprache in die Benutzersprachen-Korrespondenz-Tabelle 114 durch die Schritte 334 und möglicherweise 336 eingetragen worden ist, dann wird die Nachrichtendatei für die gewünschte Dialogsprache in Schritt 338 in den Pro-Benutzer-Privatspeicher 54 überführt. Dann wird in Schritt 338 ein Eintrag 64 in die private Verbindungstabelle 60 gemacht, indem das Nachrichtendatei-Zeigerfeld 66 derart gesetzt wird, daß die Speicherstelle, in welche die Nachrichtendatei überführt worden ist, angezeigt wird.

In Schritt 342 wird schließlich der Pro-Benutzer-Sprachcode für die gewünschte Dialogsprache zurückgeschickt.

Der Prozeß der Erweiterung der Größe der Benutzersprachen-Tabelle 110 (Fig. 11) (Schritte 336 gemäß Fig. 24, 276 gemäß Fig. 23a und 378 gemäß Fig. 21) wird nun in Verbindung mit Fig. 25 ausführlich beschrieben. Der Prozeß beginnt mit Schritt 350, in dem, wie in Feld 338, der Benutzerinformations-Tabelle 130 angezeigt ist, die maximale Anzahl von möglichen Benutzersprachen erhöht wird. Der Erhöhung der Anzahl der Benutzer-Dialogsprachen folgt der Schritt 362, in dem mehr Speicherplatz für eine größere Benutzersprachen-Tabelle 110 zugewiesen wird. Danach wird diese neue Tabelle mit Null-Worten initialisiert (Schritt 354). Nachdem die Tabelle initialisiert ist, werden die Daten der alten Benutzersprachen-Tabelle 110 in die neue Tabelle kopiert (Schritt 356). Nachdem die alten Worte in die neue Tabelle kopiert worden sind, kann in Schritt 358 der alte Speicherplatz freigegeben werden.

Wenn die Anzahl der Benutzersprachen erhöht worden ist, kann sich die Größe der privaten Verbindungstabelle 60 ebenfalls erhöhen. Deshalb wird in Schritt 360 Speicherplatz für eine neue Verbindungstabelle 60 zugewiesen, dem der Schritt 362 folgt, indem die neue Verbindungstabelle 60 mit Null-Worten initialisiert wird, der wiederum von Schritt 364 gefolgt wird, indem die alten Daten hereinkopiert werden und dann wird der Speicherplatz der alten Verbindungstabelle 60 freigegeben. In dieser Weise kann die Größe der privaten Verbindungstabelle 60 und der Benutzersprachen-Tabelle 110 dynamisch wachsen. Die Häufigkeit der Erweiterung dieser Tabellen wird dadurch minimiert, daß sie nicht nur um einen Eintrag, sondern um vier Einträge gleichzeitig vergrößert werden.

Während der Ausführung eines Anwendungsprogrammes, kann eine Nachricht zu einem Benutzer in einer dafür gewählten Benutzersprache gesendet werden. Wenn der Anwendungsprogrammierer für ein typisches Datenverarbeitungssystem eine Standardausgabe-Funktion vorgegeben hat, dann kann diese Funktion modifiziert werden, um es ihr zu ermöglichen, die richtige Nachricht in einer etablierten Dialogsprache des Benutzers zu finden. In dem ersten Schritt 390 wird festgestellt, ob es sich um ein gemeinschaftliches oder privates Anwendungsprogramm handelt. Nach dieser Feststellung wird der Freigabestatus durch eine Untersuchung der System-Freigabetabelle 90 überprüft, um festzustellen, daß das System entweder für eine gemeinschaftliche Anwendung voll freigegeben ist oder privat freigegeben ist. Diese Feststellungsschritte sind die Schritte 392 bzw. 394. Wenn das System für mehrsprachige Dialoge nicht ordnungsgemäß freigegeben ist, dann wird in Schritt 396 ein Nachrichtenzeiger mit einem Null-Wort besetzt, so als ob keine mehrsprachige Nachricht gefunden wurde. Wenn in Schritt 392 festgestellt wurde, daß das System für eine gemeinschaftliche Anwendung voll freigegeben wurde, dann wird der Prozeß mit Schritt 398 fortgesetzt, indem die Nachricht in der gemeinschaftlichen Nachrichtendatei gefunden werden soll. Der Schritt 398, nämlich die Auffindung der Nachricht, wird hiernach in Verbindung mit Fig. 27 näher beschrieben werden. Wenn in Schritt 394 festgestellt wurde, daß das System privat freigegeben ist, dann wird der Prozeß mit Schritt 397 fortgesetzt, indem die Nachricht in der privaten Nachrichtendatei gefunden werden soll. Dieses Verfahren wird in Verbindung mit Fig. 28 näher beschrieben werden.

Nach der Beendigung einer der Schritte 396, 397 oder 398, zeigt der Nachrichtenzeiger entweder Null an oder er zeigt die Speicherstelle der gewünschten Nachricht an. In Schritt 400 wird als nächstes festgestellt, ob eine Nachricht durch die Überprüfung dieses Zeigers gefunden wurde. Wenn eine Nachricht nicht gefunden wurde, d. h., wenn der Nachrichtenzeiger Null ist, dann wird das Verfahren mit Schritt 402 fortgesetzt, der die Ausgabe einer Standardnachricht behandelt. Anderenfalls wird der Prozeß mit Schritt 404 fortgesetzt, indem der gewünschte Wert für die Nachricht wird. Dieser Wert kann der Zeiger zu der Nachricht, die Nachrichten-Zeichenfolge selbst, die in einem Puffer angeordnete Nachricht oder die zu einem Benutzer an einer entsprechenden Benutzerstation 42 gesendete Nachrichten-Zeichenfolge sein.

Der Prozeß der Auffindung einer Nachricht aus einer Nachrichtendatei eines gemeinschaftlichen Anwendungsprogrammes (Schritt 408, Fig. 26) wird nun in Verbindung mit Fig. 27 ausführlich beschrieben. Der erste Schritt der Auffindung der Nachricht ist die Feststellung (Schritt 410), ob für dieses Anwendungsprogramm momentan eine Dialogsprache benutzt wird. Der Feststellungsschritt 410 enthält die Überprüfung der Felder 134 und 136 (Fig. 13), die die Sprachcodes der momentan laufenden privaten oder gemeinschaftlichen Anwendungsprogramme anzeigen. Wenn beide Felder einen Nicht-Sprachwort anzeigen, dann wird der Prozeß mit Schritt 412 fortgesetzt, indem festgestellt wird, ob die Nachrichtendatei einer Betriebssystem-Standardsprache in den Speicher des gemeinschaftlichen Anwendungsprogrammes überführt worden ist. Wenn diese Nachrichtendatei nicht in den Speicher überführt worden ist, dann wird in Schritt 414 ein Null-Nachrichtenzeiger zurückgeschickt. Anderenfalls wird der Zeiger der Nachrichtendatei aus der Verbindungstabelle 60 für gemeinschaftliche Anwendungsprogramme abgerufen und der zu der gewünschten Nachricht zeigende Zeiger wird (Schritt 416) aus dieser Nachrichtendatei abgerufen. Wenn eine momentane Dialogsprache für ein Anwendungsprogramm verfügbar ist, was in Schritt 410 festgestellt wird, dann wird in Schritt 416 der zu dieser Nachrichtendatei zeigende Zeiger und nachfolgend die Nachricht selbst abgerufen.

Der Prozeß der Auffindung einer Nachricht aus einer Nachrichtendatei für ein privates Anwendungsprogramm (Schritt 396, Fig. 26) beginnt mit Schritt 430 (Fig. 28), indem festgestellt wird, ob eine momentane Dialogsprache für ein Anwendungsprogramm benutzt wird. Dieser Feststellungsschritt 430 wird durchgeführt durch die Untersuchung der Felder 134 und 136 (Fig. 13), die den Sprachcode anzeigen, in der ein momentan laufendes Anwendungsprogramm mit einem Benutzer kommuniziert. In Schritt 434 wird durch Überprüfung der privaten Verbindungstabelle 60 als nächstes festgestellt, ob die Nachrichtendatei der momentanen Sprache für dieses Anwendungsprogramm in den Privatspeicher 54 überführt worden ist und der Eintrag 64 für dieses Anwendungsprogramm und das Nachrichtendatei-Zeigerfeld 66 (Fig. 5) wird aufgerufen. Wenn die Nachrichtendatei in den Privatspeicher überführt worden ist, dann kann der die Speicherstelle der Nachrichtendatei im Privatspeicher 54 anzeigende Zeiger durch Schritt 436 benutzt werden. Wenn festgestellt wurde (Schritt 434), daß die Nachrichtendatei nicht in den Privatspeicher überführt worden ist, dann wird in Schritt 438 festgestellt, ob in Schritt 438 zuvor nach ihr gesucht worden ist. Wenn zuvor nach ihr gesucht worden ist,

dann zeigt das Feld 66 einen Nicht-Nullwort und einen Nicht-Zeigerwort an, wie er in Schritt 330 gemäß Fig. 24 gesetzt wurde. Wenn jedoch zuvor kein Versuch durchgeführt wurde, um diese Nachricht zu finden, dann wird in Schritt 440 ein Versuch unternommen, um es in den Privatspeicher zu überführen. Dieser Schritt 440 der Überführung der Nachrichtendatei in einen Privatspeicher ist oben in Verbindung mit Fig. 24 bereits ausführlich beschrieben worden. Wenn der Versuch der Überführung erfolgreich ist, dann wird das Nachrichtendatei-Zeigerfeld 66 gesetzt und verwendet, um einen auf die gewünschte Nachricht zeigenden Zeiger in Schritt 436 zurückzuschicken, wie oben beschrieben wurde.

Wenn der Versuch 440, eine Nachrichtendatei in einen Privatspeicher zu überführen, fehlschlägt oder wenn so ein Versuch bereits zuvor durchgeführt wurde, was in Schritt 438 festgestellt wurde oder wenn in Schritt 430 festgestellt wurde, daß keine momentane Dialogsprache für dieses Anwendungsprogramm angezeigt ist, dann wird der Prozeß mit Schritt 442 fortgeführt, indem die Benutzer-Standardsprache ausgewählt wird, wie sie durch das Benutzer-Standardsprachen-Feld 100 (Fig. 9) angezeigt ist. Wenn keine Sprache gesetzt worden ist, dann wird in Schritt 444 ein Null-Zeiger zurückgeschickt. Anderenfalls wird durch Untersuchung der privaten Verbindungstabelle 60 festgestellt (Schritt 446), ob die Nachrichtendatei in der Benutzer-Standardsprache für dieses private Anwendungsprogramm in den Privatspeicher 54 überführt worden ist. Wenn ein zu dieser Nachrichtendatei zeigender Zeiger aus Feld 66 (Fig. 5) zurückgeschickt worden ist, dann kann der zu der Nachricht zeigende Zeiger in ähnlicher Weise wie der oben beschriebene Schritt 436 aus der Nachrichtendatei aufgerufen werden. Wenn festgestellt worden ist, daß die Nachrichtendatei nicht in den Privatspeicher überführt worden ist, dann wird der Prozeß mit Schritt 450 fortgesetzt, indem in einer ähnlichen Weise, wie im oben beschriebenen Schritt 438 festgestellt wird, ob zuvor nach ihm gesucht worden ist. Wenn nach dieser Nachrichtendatei zuvor gesucht worden ist, dann wird in Schritt 454 ein Null-Zeiger zurückgeschickt. Anderenfalls wird in Schritt 452 ein Versuch unternommen, um diese Nachrichtendatei in den Privatspeicher 54 zu überführen, wie dies zuvor in Verbindung mit Fig. 24 beschrieben wurde. Wenn dieser Versuch 452 erfolgreich ist, dann kann ein zu der gewünschten Nachricht zeigender Zeiger unter Verwendung einer Speicherstelle der Nachrichtendatei im Privatspeicher wie in Schritt 448 aufgerufen werden. Wenn der Versuch 452 fehlschlägt, dann wird in Schritt 454 ein Null-Zeiger zurückgeschickt.

Die Fähigkeit des Systems auf natürliche Spracheingaben von einem Benutzer unter Verwendung einer Benutzerstation 42 zu antworten, kann ebenso geschaffen werden. Standardeingaben-Verarbeitungsfunktionen können derart modifiziert werden, daß sie die im folgenden in Verbindung mit Fig. 29 beschriebenen Schritte aufweisen. Für diesen Prozeß enthält eine Nachrichtendatei ebenso Eingaben von einem Benutzer in natürlicher Sprache, z. B. "Ja", "Nein" und "Verlasse". Diese Eingaben Nachrichten können in die Nachrichtendatei zusammen mit Ausgaben Nachrichten aufgenommen werden. Der erste Schritt der Verarbeitung von Eingaben in natürlicher Sprache enthält den Schritt 460, in dem festgestellt wird, ob die mehrsprachigen Eigenschaften des Systems gemäß dem Typ des Anwendungsprogrammes (d. h. gemeinschaftlich oder privat) in geeigneter Weise freigegeben wurden. Wenn diese Eigenschaft nicht fr i-

gegeben ist, dann wird in Schritt 462 ein Falsch-Wert zurückgeschickt. Der Typ des Programms kann durch Untersuchung der Speicherstelle, in welcher es gespeichert ist, festgestellt werden. Der Freigabestatus kann aus der System-Freigab-Tabelle 90 (Fig. 8) festgestellt werden; und insbesondere in Feld 92 für private Anwendungen und in Feld 96 für gemeinschaftliche Anmeldungen. Wenn festgestellt worden ist, daß es sich bei dem Anwendungsprogramm um ein gemeinschaftliches Anwendungsprogramm handelt und wenn das System voll freigegeben ist, dann wird der Prozeß mit Schritt 466 fortgeführt, in dem die erste Nachricht, die mit der Eingabe des Benutzers verglichen werden soll, gesetzt wird. Ein Anwendungsprogrammierer würde normalerweise eine Eingabe von einem Benutzer erwarten, die einer abschließenden Liste von möglichen Eingaben Nachrichten entspricht. Der Anwendungsprogrammierer kann dann die Eingabe mit jeder dieser erwarteten Antworten vergleichen. Selbst ein Argument "N" an die erste dieser erwarteten Nachrichten. Der Prozeß wird mit Schritt 468 fortgeführt, in dem diese erwartete Nachricht in einer Nachrichtendatei gefunden werden soll. Wenn die Nachricht nicht gefunden worden ist, dann wird in den Schritten 474 und 468 nach der nächsten erwarteten Nachricht gesucht. Wenn eine Nachricht gefunden worden ist, dann wird diese Nachricht in Schritt 476 mit der Eingabe des Benutzers verglichen.

Der Schritt 476 der Abgleichung der aufgerufenen Nachricht mit der Eingabe des Benutzers kann in einer Vielzahl von Wegen durchgeführt werden; ein Abgleich kann partiell, ähnlich oder exakt sein. Wenn die Nachricht mit einer Eingabe des Benutzers übereinstimmt, dann kann in Schritt 478 die Eingabe, die Nachricht oder ein Wahr-Wert zurückgeschickt werden. Es ist möglich, diese vergleichende Suche fortzusetzen, um andere Abgleiche mit Nachrichten zu finden und um lieber eine Liste von partiellen Abgleichen als einen einzelnen und ersten Abgleich zurückzuschicken. Wenn die Eingabe des Benutzers nicht zu der aufgerufenen, erwarteten Nachricht paßt, dann wird in Schritt 474 die nächste erwartete Nachricht aufgerufen und der Prozeß wird mit dem bereits oben beschriebenen Schritt 468 fortgesetzt. Wenn es keine verbleibenden, erwarteten Nachrichten mehr gibt und wenn keine gefunden wurde, die auf die Eingabe des Benutzers paßt, dann wird in Schritt 480 ein Falsch-Wert zurückgeschickt. Dieser Schritt 480 kann alternativ die Liste von zuvor gefundenen partiellen Abgleichen zurückschicken. In diesem Falle wird die Liste mit einem Falsch-Wert initialisiert.

Für den Fall einer Benutzereingabe für ein privates Anwendungsprogramm beginnt der Prozeß des Abgleichs der Nachricht mit Schritt 482, in dem die erste erwartete Nachricht ausgewählt wird, vorausgesetzt, das System ist zumindest privat freigegeben. Der Prozeß wird mit Schritt 484 fortgesetzt, indem die Nachricht in der Nachrichtendatei für dieses private Anwendungsprogramm gefunden werden soll. Die für diesen Prozeß der Abgleichung der Eingabe mit einer Nachricht verbleibenden Schritte 490, 492, 494 und 496 sind den Schritten 476, 478, 474 und 480 zur Auffindung einer Nachricht für ein gemeinschaftliches Anwendungsprogramm ähnlich. In jedem Fall wird eine Eingabe des Benutzers in natürlicher Sprache mit einer Gruppe von erwarteten Eingaben verglichen, so daß der volle Dialog zwischen einem Benutzer und dem Datenverarbeitungssystem in einer natürlichen Sprache freigegeben ist.

Zu gegebener Zeit kann es wünschenswert sein, ein

Anwendungsprogramm oder ein Wörterbuch und ihre Nachrichtendateien aus dem Speicher oder aus dem Gebrauch zu entfernen (Fig. 30). Der erste Schritt 500 des Entfernens einer Anwendung vom Gebrauch ist die Identifikation der Speicherstelle des Prozeßcodes oder anderer Kennzeichnungen des Anwendungsprogrammes. Die in Schritt 500 gefundene Kennzeichnung wird in Schritt 502 als ein Suchindex für die Verbindungstabelle 60 verwendet und der Eintrag 64 wird aufgerufen. Für die im Feld 66 angezeigte, korrespondierende Nachrichtendatei wird die Zuordnung zurückgenommen und der Eintrag 64 wird aus der Verbindungstabelle 60 entfernt (Schritt 504). Für das Anwendungsprogramm oder das Wörterbuch selbst kann die Zuordnung dann zurückgenommen werden (Schritt 505).

Wenn ein Anwendungsprogramm ein verschachteltes Anwendungsprogramm ist, d. h., wenn es von einem anderen Anwendungsprogramm aufgerufen wurde, dann wird die Programmebene wie in Fig. 31 dargestellt wieder hergestellt. Das durch den Benutzer verschachtelte Sprachen-Stapelregister 102 wird verwendet, um einen Eintrag 104 für das Anwendungsprogramm aufzurufen, welches zuvor gelaufen ist. Dieser Eintrag 104 (Fig. 10) wird verwendet, um die Felder 134 und 136 (Fig. 13) der Benutzer-Informations-Tabelle 130 zu setzen, die die Dialogsprache anzeigen, in der das momentane Anwendungsprogramm läuft (Schritt 506). Der auf das verschachtelte Sprachen-Stapel-Register (Fig. 30) zeigende Zeiger 132 wird dann auf den nächsten Eintrag 104 des Stapelregisters 102 zurückgesetzt (Schritt 508).

Für ein mehrsprachiges Datenverarbeitungssystem kann es ebenso wünschenswert sein, einen Benutzer zu ermächtigen oder eine Liste von Dialogsprachen zu erhalten, die für ein Anwendungsprogramm verfügbar sind. Zwei Arten von Listen sind möglich. Die erste dieser Listen enthält die Sprachen für ein Anwendungsprogramm, für welches Nachrichtendateien in den Privat- oder Gemeinschaftsspeicher überführt oder geladen worden sind. Der andere Typ von Liste ist eine Liste der Dialogsprachen-Nachrichtendateien, die für ein gegebenes Anwendungsprogramm auf einer Festplatte existieren. Das erstgenannte ist hauptsächlich für Gemeinschaftsanwendungen wünschenswert. Das zweite ist vorzugsweise für private Anwendungen, deren Nachrichtendateien nicht alle in den Privatspeicher überführt wurden.

Der erste Listentyp wird nun in Verbindung mit Fig. 32 beschrieben. Diese Konstruktion beginnt mit Schritt 510, in dem festgestellt wird, ob der mehrsprachige Dialog auf dem System für den Typ des Anwendungsprogrammes in geeigneter Weise freigegeben ist. Wenn es nicht freigegeben ist, dann wird der Benutzer benachrichtigt, daß für dieses Anwendungsprogramm keine Sprachen verfügbar sind (Schritt 512). Anderenfalls wird ein Argument, z. B. "N" zu der maximalen Anzahl von Sprachen gesetzt (Schritt 514), wie sie in Feld 72 der System-Informations-Tabelle 70 (Fig. 6) gefunden wurde (Schritt 514). Ein zweites Argument z. B. "L" wird zu "Null" gesetzt (Schritt 516). Das zweite Argument wird dann in Schritt 518 mit dem ersten Argument verglichen. Wenn das zweite Argument "L" größer ist als das erste Argument "N", dann wird dadurch angezeigt, daß die Anzahl der gesuchten Sprachen größer ist als die Anzahl der maximalen Sprachen und der Prozeß endet dann (Schritt 520). Anderenfalls wird festgestellt, ob eine Sprache entsprechend dem zweiten Argument "L" aufweisende Nachrichtendatei einer Dialogsprache durch Untersuchung der geeigneten Verbin-

dungstabelle 60 und durch Untersuchung des Nachrichtendatei-Zeigerfeldes 66 für das Anwendungsprogramm 62 und die Sprache 68 (Fig. 5) installiert wurde. Wenn in Schritt 522 herausgefunden wurde, daß die Nachrichtendatei der Sprache geladen wurde, dann wird in Schritt 524 eine Nachricht in einer geeigneten Dialogsprache zu dem Benutzer übersandt, die anzeigt, daß die dem zweiten Argument "L" korrespondierende Sprache verfügbar ist. Wenn die Sprache nicht installiert ist oder wenn das zweite Argument "L" in Schritt 526 vergrößert wurde, nachdem die Nachricht, die anzeigt, daß es installiert ist, zu dem Benutzer ausgesandt wurde, dann wird der Prozeß fortfahren mit dem Vergleich von Schritt 518, wie es bereits oben beschrieben ist. In dieser Weise erhält der Benutzer an der Benutzerstation einen Strom von Ausgabenachrichten, die anzeigen, welche Sprachen für das ausgewählte Anwendungsprogramm verfügbar sind.

Der Aufbau des zweiten Listentyps von verfügbaren Dialogsprachen wird nun in Verbindung mit Fig. 33 beschrieben. Der erste Schritt dieses Aufbaus ist Schritt 530, in dem festgestellt wird, ob ein mehrsprachiger Dialog für das System bereits freigegeben worden ist. Wenn es noch nicht freigegeben worden ist, dann wird in Schritt 532 eine Nachricht an den Benutzer ausgesandt, die anzeigt, daß keine Sprachen verfügbar sind. Anderenfalls wird in Schritt 534 der Speicher des Datenverarbeitungssystems abgesucht, um die Nachrichtendateien eines gegebenen Anwendungsprogramms zu finden. Diese Suche wird durchgeführt unter Verwendung eines Dateinamens, der auf dem Anwendungsprogramm und dem Dateitypen einer Nachrichtendatei basiert, wobei jeder Sprachename akzeptiert wird, der sich in der Nachrichtendatei befindet. Für jede in Schritt 534 gefundene Nachrichtendatei wird der dieser Nachrichtendatei entsprechende Sprachennamen zum Teil einer Nachricht gemacht, die dem Benutzer übersandt wird, um anzuzeigen, daß diese Dialogsprache für das gegebene Anwendungsprogramm verfügbar ist (Schritt 536).

Mit der oben beschriebenen Ausführungsform kann ein Benutzer eine gewünschte Dialogsprache spezifizieren, damit sodann der Dialog mit dem Datenverarbeitungssystem in dieser Sprache durchgeführt werden kann, wenn diese Sprache für ein gegebenes Anwendungsprogramm verfügbar ist. Solch ein System kann in einer Vielzahl von mehrsprachigen Umgebungen besonders nützlich sein.

Nachdem nun eine Ausführungsform der Erfindung beschrieben wurde, sollte es dem Fachmann dieses Gebiets bewußt sein, daß die obige Beschreibung keinen beschränkenden, sondern illustrativen Charakter hat, die hier nur beispielhaft abgegeben wurde. Eine Vielzahl von anderen Ausführungsformen und Abwandlungen hiervon werden als unter den Schutzzumfang der Erfindung, wie sie durch die beigefügten Ansprüche und deren Äquivalente definiert wird, fallend betrachtet.

Patentansprüche

1. Ein Datenverarbeitungssystem, welches einen Benutzer-Dialog in mehreren natürlichen Sprachen ermöglicht, enthält:
einen Prozessor zur Ausführung eines Anwendungsprogrammes;
einen Speicher zur Speicherung wenigstens einer dem Anwendungsprogramm zugewiesenen Nachrichtendatei, wobei eine Nachrichtendatei Nachrichten in wenigstens einer natürlichen Sprache

enthält;
 Mittel zum Auswählen einer natürlichen Sprache;
 Mittel zum Abrufen einer Nachricht von der Nachrichtendatei, wobei die Nachrichtendatei Nachrichten in der ausgewählten, natürlichen Sprache enthält;
 eine Verbindungstabelle zur Speicherung wenigstens einer Verbindung zwischen dem Anwendungsprogramm und der wenigstens einen Nachrichtendatei, wobei eine Verbindung eines Anwendungsprogrammes Worte speichert, die auf die wenigstens eine natürliche Sprache und die Lage der Nachrichtendatei im Speicher hinweisen;
 wobei die Mittel zum Abrufen Mittel zum Suchen enthalten, die die Verbindungstabelle nach der Verbindung des Anwendungsprogrammes absuchen und die die ausgewählte, natürliche Sprache nach einem Wort absuchen, das auf die Lage der Nachrichtendatei der ausgewählten Sprache hinweist; und
 Mittel zur Speicherung einer aktuellen Dialogsprache und Mittel zur Speicherung einer Standardsprache, wobei die Mittel zum Auswählen einer natürlichen Sprache zwischen der aktuellen Dialogsprache und der Standardsprache auswählen.
 2. Ein Datenverarbeitungssystem nach Anspruch 1, wobei jede Nachrichtendatei nur Nachrichten in einer natürlichen Sprache enthält.
 3. Ein Datenverarbeitungssystem nach Anspruch 1, wobei der Prozessor Anwendungsprogramme verschachteln kann und welches weiterhin enthält:
 Mittel zur Speicherung von Sprachinformationen für verschachtelte Anwendungsprogramme; und
 Mittel, die auf einen Aufruf eines anderen Anwendungsprogrammes durch das Anwendungsprogramm antworten um die gespeicherte, momentane Dialogsprache in den Mitteln zur Speicherung verschachtelter Sprachinformationen zu sichern und um die momentane Dialogsprache als die gewünschte Dialogsprache zu setzen.
 4. Ein Datenverarbeitungssystem nach Anspruch 3, wobei der Prozessor durch ein Betriebssystem gesteuert wird, welches als Antwort auf Anweisungen des Anwendungsprogrammes arbeitet und welches weiterhin enthält:
 Mittel zur Ladung der Nachrichtendatei der gewünschten Dialogsprache des aufgerufenen Anwendungsprogrammes in den Speicher; und
 Mittel im Betriebssystem zum Aufbau einer Verbindung in der Verbindungstabelle als Antwort auf die Mittel zur Ladung einer Nachrichtendatei.
 5. Ein Datenverarbeitungssystem nach Anspruch 4, enthält weiterhin Mittel zum Auswählen der gewünschten Dialogsprache aus der momentanen Dialogsprache und einer Standard-Dialog-Sprache.
 6. Ein Datenverarbeitungssystem nach Anspruch 5, wobei: der Befehl, ein anderes Anwendungsprogramm aufzurufen, eine gewollte Dialogsprache enthält; und die Mittel zur Auswahl der gewünschten Dialogsprache auch die gewollte Dialogsprache auswählen können.
 7. Ein Datenverarbeitungssystem nach Anspruch 6, welches weiterhin enthält:
 Mittel, die auf eine Beendigung des aufgerufenen Anwendungsprogrammes antworten um die momentane Dialogsprache als eine vorherige Dialogsprache in den Mitteln zur Speicherung verschachtelter Sprachinformationen zu setzen; und

Mittel zur Beseitigung der vorherigen Dialogsprache aus den Mitteln zur Speicherung verschachtelter Sprachinformationen.

8. Ein Datenverarbeitungssystem nach Anspruch 7, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen ein Stapelregister ist.

9. Ein Datenverarbeitungssystem nach Anspruch 8, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen eine verbundene Liste ist.

10. Ein Datenverarbeitungssystem nach Anspruch 1, wobei das Mittel zum Auswählen einer natürlichen Sprache auf einen Befehl eines Benutzers antwortet, der ein Anwendungsprogramm unter Verwendung einer gewünschten Dialogsprache aufruft, um die momentane Dialogsprache als die gewünschte Dialogsprache zu setzen, falls eine Nachrichtendatei, die Nachrichten in der gewünschten Dialogsprache enthält, verfügbar ist und um die momentane Dialogsprache als eine Standard-Dialogsprache zu setzen, falls die gewünschte Dialogsprache nicht verfügbar ist.

11. Datenverarbeitungssystem nach Anspruch 10, wobei der Prozessor durch ein Betriebssystem gesteuert wird, welches als Antwort auf Instruktionen des Anwendungsprogrammes arbeitet und welches weiterhin enthält:

Mittel zum Laden der Nachrichtendatei der ausgewählten, natürlichen Sprache des aufgerufenen Anwendungsprogrammes in den Speicher; und
 Mittel im Betriebssystem zum Aufbau einer Verbindung in der Verbindungstabelle als Antwort auf die Mittel zum Laden einer Nachrichtendatei.

12. Ein Datenverarbeitungssystem, welches einen Benutzer-Dialog in mehreren natürlichen Sprachen schafft, enthält:

einen Prozessor, der durch ein Betriebssystem gesteuert wird, welches als Antwort auf Instruktionen eines Anwendungsprogrammes arbeitet;

einen Speicher zur Speicherung wenigstens einer dem Anwendungsprogramm zugehörigen Nachrichtendatei, wobei jede Nachrichtendatei Nachrichten in einer natürlichen Sprache enthält;
 eine Verbindungstabelle zur Speicherung wenigstens einer Verbindung zwischen dem Anwendungsprogramm und der wenigstens einen Nachrichtendatei, wobei jede Verbindung einen natürlichen Sprache anzeigenden Wert und einen die Nachrichtendatei des Anwendungsprogrammes, die Nachrichten in der natürlichen Sprache enthält, anzeigenden Zeiger speichert;

Mittel zum Laden einer Nachrichtendatei des Anwendungsprogrammes in den Speicher; und

Mittel im Betriebssystem, die auf die Mittel zum Laden antworten, um eine Verbindung in der Verbindungstabelle zwischen dem Anwendungsprogramm, dem auf die Nachrichtendatei zeigenden Zeiger und der natürlichen Sprache der Nachrichten in der Nachrichtendatei aufzubauen.

13. Ein Datenverarbeitungssystem nach Anspruch 12, welches weiterhin enthält:

Mittel, die auf einen Befehl zur Anweisung des Prozessors antworten, um das Anwendungsprogramm auszuführen;

Mittel zum Auswählen einer Dialogsprache; und
 wobei die Mittel zum Laden einer Nachrichtendatei die zugehörige Nachrichtendatei der ausgewählten Dialogsprache als Antwort auf die Mittel zur Anweisung laden.

14. Ein Datenverarbeitungssystem nach Anspruch 13, welches weiterhin Mittel zur Speicherung einer Standardsprache enthält und wobei der Befehl zur Anweisung des Prozessors eine gewollte Dialogsprache enthält und wobei die Mittel zum Auswählen einer Dialogsprache diese aus der gewollten Dialogsprache und der Standard-Dialogsprache auswählt. 5

15. Ein Datenverarbeitungssystem nach Anspruch 14, wobei der Befehl zur Aufrufung des Anwendungsprogrammes durch ein anderes Anwendungsprogramm ausgegeben wird und welches weiterhin enthält: 10

Mittel zur Speicherung einer momentanen Dialogsprache; 15

Mittel zur Speicherung verschachtelter Sprachinformationen; und

Mittel, die auf den Befehl zur Aufrufung des Anwendungsprogrammes antworten, um die momentane Dialogsprache als die ausgewählte Dialogsprache 20

setzen und zur Einsparung einer vorherigen, momentanen Dialogsprache in den Mitteln zur Speicherung verschachtelter Sprachinformationen.

16. Ein Datenverarbeitungssystem nach Anspruch 15, wobei die Mittel zur Auswahl der Dialogsprache auch die vorherige, momentane Dialogsprache auswählen können. 25

17. Ein Datenverarbeitungssystem nach Anspruch 16, welches weiterhin enthält: 30

Mittel, die auf einen Befehl des Anwendungsprogrammes antworten, um eine Nachricht zur Auswahl einer natürlichen Sprache abzurufen und zur Abrufung dieser Nachricht aus der Nachrichtendatei, die Nachrichten in der ausgewählten, natürlichen Sprache enthält. 35

18. Ein Datenverarbeitungssystem nach Anspruch 17, wobei die Mittel zur Auswahl einer natürlichen Sprache aus der momentanen Dialogsprache und der Standarddialogsprache auswählen.

19. Ein Datenverarbeitungssystem nach Anspruch 18, welches weiterhin enthält: 40

Mittel, die auf eine Beendigung des Anwendungsprogrammes antworten, um die momentane Dialogsprache als die vorherige Dialogsprache aus den Mitteln zur Speicherung verschachtelter Sprachinformationen zu setzen; und 45

Mittel zum Entfernen der vorherigen Dialogsprache aus den Mitteln zur Speicherung verschachtelter Sprachinformationen.

20. Ein Datenverarbeitungssystem nach Anspruch 19, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen ein Stapelregister ist. 50

21. Ein Datenverarbeitungssystem nach Anspruch 20, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen eine verbundene Liste ist. 55

22. Ein Datenverarbeitungssystem um einen Benutzer-Dialog in mehreren natürlichen Sprachen zu schaffen, enthält: 60

einen Prozessor zur Ausführung eines Anwendungsprogrammes, welches Anwendungsprogramme verschachteln kann;

einen Speicher zur Speicherung wenigstens einer Nachrichtendatei; 65

Mittel zur Speicherung einer momentanen Dialogsprache;

eine verschachtelte Sprach-Informationen-Tabelle zur Speicherung von Sprachinformationen von verschachtelten Anwendungsprogrammen; und

Mittel zur Aufrufung anderer Anwendungsprogramme durch das Anwendungsprogramm, um die momentane Dialogsprache in den Mitteln zur Speicherung verschachtelter Sprachinformationen zu sichern und um die momentane Dialogsprache als die gewünschte Dialogsprache zu setzen. n.

23. Datenverarbeitungssystem nach Anspruch 22, welches weiterhin Mittel zur Speicherung einer Dialogsprache enthält, wobei die Mittel zum Setzen der momentanen Dialogsprache als eine gewünschte Dialogsprache Mittel zur Auswahl der gewünschten Dialogsprache aus der momentanen Dialogsprache und der Standard-Dialogsprache enthalten.

24. Ein Datenverarbeitungssystem nach Anspruch 23, wobei 70

der Befehl, ein anderes Anwendungsprogramm aufzurufen, eine gewollte Dialogsprache enthält; und

das Mittel zur Auswahl der gewünschten Dialogsprache auch die gewollte Dialogsprache auswählen kann.

25. Ein Datenverarbeitungssystem nach Anspruch 24, welches weiterhin enthält: 75

Mittel, die auf einen Befehl des Anwendungsprogrammes antworten, um eine Nachricht zum Auswählen einer natürlichen Sprache aus der momentanen Dialogsprache und der Standard-Dialogsprache abzurufen und zur Abrufung dieser Nachricht aus der Nachrichtendatei, die Nachrichten in der ausgewählten, natürlichen Sprache enthält.

26. Ein Datenverarbeitungssystem nach Anspruch 25, welches weiterhin enthält: 80

Mittel, die auf eine Beendigung des anderen Anwendungsprogrammes antworten, um die momentane Dialogsprache als die vorherige Dialogsprache aus den Mitteln zur Speicherung verschachtelter Sprachinformationen zu setzen; und

Mittel zur Entfernung der vorherigen Dialogsprache aus den Mitteln zur Speicherung verschachtelter Sprachinformationen.

27. Ein Datenverarbeitungssystem nach Anspruch 26, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen ein Stapelregister ist.

28. Ein Datenverarbeitungssystem nach Anspruch 26, wobei das Mittel zur Speicherung verschachtelter Sprachinformationen eine verbundene Liste ist.

29. Ein Verfahren zur Schaffung eines Benutzerdialogs in mehreren natürlichen Sprachen, enthält die folgenden Schritte: 85

Schaffung eines Anwendungsprogrammes und eines Speichers zur Speicherung wenigstens einer dem Anwendungsprogramm zugehörigen Nachrichtendatei, wobei jede Nachrichtendatei Nachrichten in einer natürlichen Sprache enthält; Auswahl einer natürlichen Sprache als Antwort auf einen Befehl, eine Nachricht durch das Anwendungsprogramm abzurufen; und

Abrufung der Nachricht von der Nachrichtendatei, die Nachrichten in der ausgewählten, natürlichen Sprache enthält.

30. Ein Verfahren nach Anspruch 29, welches weiterhin den Schritt der Schaffung einer Verbindungstabelle zur Speicherung wenigstens einer Verbindung zwischen dem Anwendungsprogramm und der wenigstens einen Nachrichtendatei enthält, wobei eine Verbindung ein eine natürliche Sprache anzuzeigendes Wort und einen die Nachrichtenda-

tei, die Nachrichten in der natürlichen Sprache enthält, anzeigenden Zeiger speichert; und wobei die Schritte der Auswahl und der Abrufung den Schritt des Absuchens der Verbindungstabelle des Anwendungsprogramms und der ausgewählten natürlichen Sprache und die Abrufung des auf die Nachrichtendatei der ausgewählten natürlichen Sprache zeigenden Zeigers enthält.

31. Ein Verfahren nach Anspruch 30, welches weiterhin die Schritte enthält:

Speicherung einer momentanen Dialogsprache; und

Speicherung einer Standardsprache, wobei der Schritt der Auswahl einer natürlichen Sprache die Auswahl aus der momentanen Dialogsprache und der Standardsprache enthält.

32. Ein Verfahren nach Anspruch 29, welches weiterhin die Schritte enthält:

Speicherung einer momentanen Dialogsprache;

Speicherung verschachtelter Sprachinformationen; und

Setzen der momentanen Dialogsprache als eine gewünschte Dialogsprache als Antwort auf einen Befehl des Anwendungsprogramms um ein anderes Anwendungsprogramm aufzurufen und Sicherung einer vorherigen, momentanen Dialogsprache mit den gespeicherten verschachtelten Sprachinformationen.

33. Ein Verfahren nach Anspruch 32, welches weiterhin Mittel zur Ladung der Nachrichtendatei der momentanen Dialogsprache des aufgerufenen Anwendungsprogramms in einen Speicher enthält; und Aufbau einer Verbindung in der Verbindungstabelle als Antwort auf den Schritt der Ladung einer Nachrichtendatei.

34. Ein Verfahren nach Anspruch 33, welches weiterhin den Schritt der Auswahl der gewünschten Dialogsprache aus der momentanen Dialogsprache und einer Standarddialogsprache enthält.

35. Ein Verfahren nach Anspruch 34, wobei der Schritt der Auswahl der gewünschten Dialogsprache den Schritt der Ladung der Nachrichtendatei der gewünschten Dialogsprache des aufgerufenen Anwendungsprogramms in einen Speicher und weiterhin den Schritt des Aufbaus einer Verbindung in der Verbindungstabelle im Anschluß an den Schritt der Ladung der Nachrichtendatei enthält.

36. Ein Verfahren nach Anspruch 35, wobei: der Befehl, ein anderes Anwendungsprogramm aufzurufen eine gewollte Dialogsprache enthält; und

der Schritt der Auswahl der gewünschten Dialogsprache weiterhin die Auswahl der gewollten Dialogsprache enthält.

37. Ein Verfahren nach Anspruch 36, welches weiterhin die Schritte enthält:

Setzen der momentanen Dialogsprache als Antwort auf eine Beendigung des aufgerufenen Anwendungsprogramms als eine vorherige Dialogsprache aus den gespeicherten, verschachtelten Sprachinformationen; und

Entfernung der vorherigen Dialogsprache aus den gespeicherten, verschachtelten Sprachinformationen.

38. Ein Verfahren nach Anspruch 37, wobei die gespeicherte, verschachtelte Sprachinformation in einem Stapelregister gespeichert ist.

39. Ein Verfahren nach Anspruch 38, wobei die verschachtelte Sprachinformation in einer verbundenen Liste gespeichert ist.

40. Ein Verfahren nach Anspruch 39, welches weiterhin enthält:

Speicherung einer momentanen Dialogsprache; und Setzen der momentanen Dialogsprache als Antwort auf einen Befehl eines Benutzers ein Anwendungsprogramm aufzurufen unter Verwendung einer gewünschten Dialogsprache als die gewünschte Dialogsprache, wenn eine Nachrichtendatei, die Nachrichten in der gewünschten Dialogsprache enthält, verfügbar ist; und

Setzen der momentanen Dialogsprache als eine Standarddialogsprache, falls die gewünschte Dialogsprache nicht verfügbar ist.

41. Ein Verfahren nach Anspruch 40, welches weiterhin die Schritte enthält:

Ladung der Nachrichtendatei der momentanen Dialogsprache des aufgerufenen Anwendungsprogramms; und

Aufbau einer Verbindung in der Verbindungstabelle für die geladene Nachrichtendatei und die momentane Dialogsprache.

42. Ein Verfahren zur Schaffung eines Benutzer-Dialogs in mehreren natürlichen Sprachen mit den Schritten:

Bereitstellen eines Anwendungsprogramms und wenigstens einer dem Anwendungsprogramm zugehörigen Nachrichtendatei, wobei jede Nachrichtendatei Nachrichten in einer natürlichen Sprache enthält;

Schaffung einer Verbindungstabelle zur Speicherung wenigstens einer Verbindung zwischen dem Anwendungsprogramm und der wenigstens einen Nachrichtendatei, wobei jede Verbindung einen eine natürliche Sprache anzeigenden Wert und einen die Nachrichtendatei des Anwendungsprogramms, die Nachrichten in einer natürlichen Sprache enthält, anzeigenden Zeiger speichert;

Laden einer Nachrichtendatei des Anwendungsprogramms in den Speicher; und

Aufbau einer Verbindung in der Verbindungstabelle als Antwort auf ein Laden einer Nachrichtendatei zwischen dem Anwendungsprogramm, dem die Nachrichtendatei anzeigenden Zeiger und der natürlichen Sprache der Nachrichten in der Nachrichtendatei.

43. Ein Verfahren nach Anspruch 42, welches weiterhin die Schritte enthält:

Anweisung eines Prozessors, um das Anwendungsprogramm auszuführen;

Auswahl einer Dialogsprache; und

wobei der Schritt der Ladung einer Nachrichtendatei das Laden der zugehörigen Nachrichtendatei als Antwort einer Anweisung an den Prozessor der ausgewählten Dialogsprache enthält.

44. Ein Verfahren nach Anspruch 43, welches weiterhin Mittel zur Speicherung einer Standardsprache enthält und wobei der Befehl zur Anweisung des Prozessors eine gewollte Dialogsprache enthält, die Mittel zur Auswahl einer Dialogsprache aus der gewollten Dialogsprache und der Standard-Dialogsprache enthält.

45. Ein Verfahren nach Anspruch 44, wobei der Befehl zum Aufrufen des Anwendungsprogramms von einem anderen Anwendungsprogramm ausgegeben wird und welches weiterhin die Schritte ent-

hält:

Speicherung einer momentanen Dialogsprache;
Speicherung verschlüsselter Sprachinformationen;
und

Setzen der momentanen Dialogsprache als die aus- 5
gewählte Dialogsprache; und
Speicherung einer vorherigen, momentanen Dia-
logsprache als Antwort auf einen Aufruf des An-
wendungsprogramms mit gespeicherten, ver-
schachtelten Sprachinformationen. 10

46. Ein Verfahren nach Anspruch 45, wobei der
Schritt der Auswahl der Dialogsprache auch aus
der vorherigen, momentanen Dialogsprache aus-
wählen kann. 15

47. Verfahren nach Anspruch 46, welches weiterhin 15
die Schritte enthält:

Auswahl einer natürlichen Sprache als Antwort auf
einen Befehl des Anwendungsprogramms, eine
Nachricht aufzurufen, und
Aufrufen einer Nachricht aus der Nachrichtendatei, 20
welche Nachrichten in der ausgewählten Sprache
enthält.

48. Ein Verfahren nach Anspruch 47, wobei der
Schritt der Auswahl einer natürlichen Sprache eine
Auswahl aus der momentanen Dialogsprache und 25
der Standard-Dialogsprache enthält.

49. Ein Verfahren nach Anspruch 48, welches wei-
terhin die Schritte enthält:

Setzen der momentanen Dialogsprache mit der
vorherigen Dialogsprache in den gespeicherten, 30
verschachtelten Sprachinformationen als Antwort
auf eine Beendigung des Anwendungsprogramms;
und

Entfernen der vorherigen Dialogsprache aus den
gespeicherten, verschachtelten Sprachinformatio- 35
nen.

50. Ein Verfahren nach Anspruch 49, wobei die ge-
speicherte, verschachtelte Sprachinformation ein
gespeichertes Stapelregister ist.

51. Ein Verfahren nach Anspruch 50, wobei die ver- 40
schachtelte Sprachinformation in einer verbun-
denen Liste gespeichert ist.

52. Ein Verfahren zur Schaffung eines Bediener-
Dialogs in mehreren natürlichen Sprachen enthält
die Schritte: 45

Schaffung eines Anwendungsprogramms und we-
nigstens einer Nachrichtendatei;
Speicherung einer momentanen Dialogsprache;
Speicherung verschlüsselter Sprachinformationen;
Setzen der momentanen Dialogsprache als eine ge- 50
wünschte Dialogsprache als Antwort auf einen Be-
fehl des Anwendungsprogramm, ein anderes An-
wendungsprogramm aufzurufen und
Sicherung einer vorherigen momentanen Dialog-
sprache mit den gespeicherten, verschachtelten 55
Sprachinformationen.

53. Ein Verfahren nach Anspruch 52, welches wei-
terhin den Schritt der Speicherung einer Standard-
sprache enthält und wobei der Schritt des Setzens
der momentanen Dialogsprache als eine gewünsch- 60
te Dialogsprache die Auswahl der gewünschten
Dialogsprach aus der momentanen Dialogsprache
und der Standard-Dialogsprach enthält.

54. Ein Verfahren nach Anspruch 53, wobei:
Der Befehl, ein anderes Anwendungsprogramm 65
aufzurufen, eine gewollte Dialogsprache enthält;
und
der Schritt der Auswahl der gewünschten Dialog-

sprache auch die Auswahl der gewollten Dialog-
sprache enthält.

55. Ein Verfahren nach Anspruch 54, welches wei-
terhin enthält:

Auswahl einer natürlichen Sprache aus der mo-
mentanen Dialogsprache und der Standard-Dialog-
sprache als Antwort auf einen Befehl des Anw
ndungsprogramms eine Nachricht abzurufen und
zum Abrufen der Nachricht aus der Nachrichten-
datei, die Nachrichten in der ausgewählten natürli-
chen Sprache enthält.

56. Ein Verfahren nach Anspruch 55, welches wei-
terhin enthält:

Setzen der momentanen Dialogsprache als Ant-
wort auf eine Beendigung des anderen Anwen-
dungsprogramms als die vorherige Dialogsprache,
die aus der gespeicherten, verschachtelten Sprach-
information genommen ist; und
Entfernen der vorherigen Dialogsprache aus den
gespeicherten, verschachtelten Sprachinformatio-
nen.

57. Ein Verfahren nach Anspruch 56, wobei die Mit-
tel zur Speicherung verschachtelter Sprachinfor-
mationen in einem Stapelregister gespeichert sind.

58. Ein Verfahren nach Anspruch 56, wobei die ver-
schachtelten Sprachinformationen in einer verbun-
denen Liste gespeichert sind.

Hierzu 28 Seite(n) Zeichnungen

- Leerseite -

THIS PAGE BLANK (USPTO)

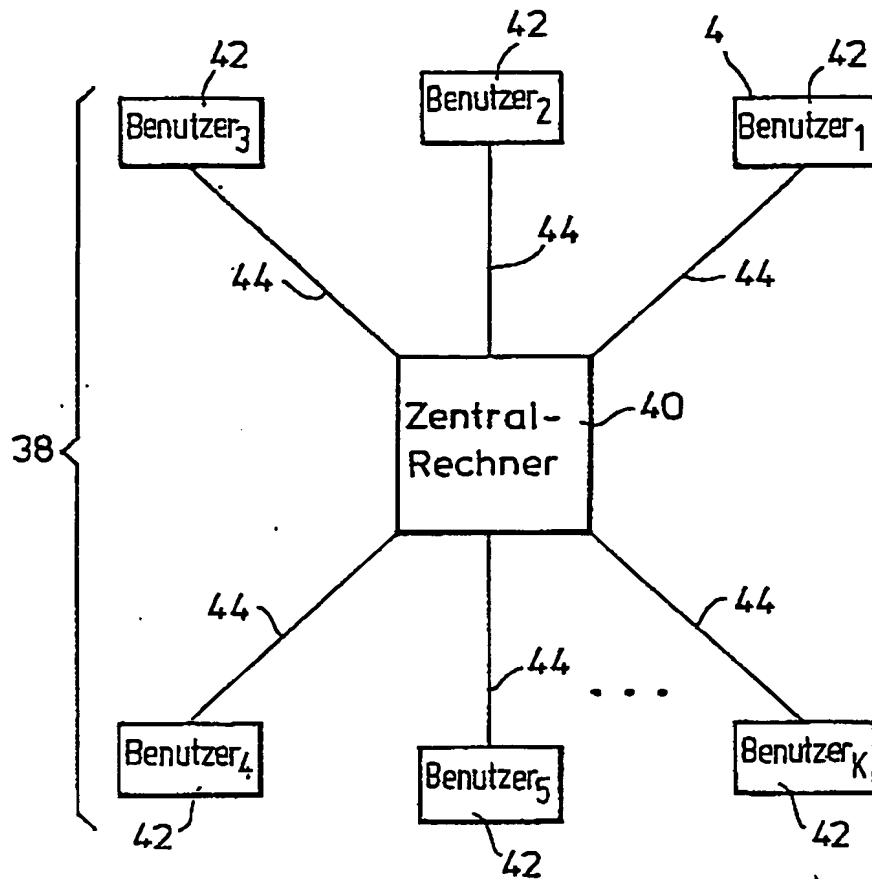


Fig. 1

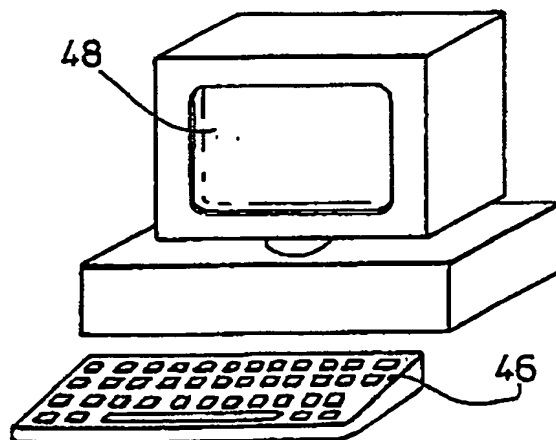


Fig. 2

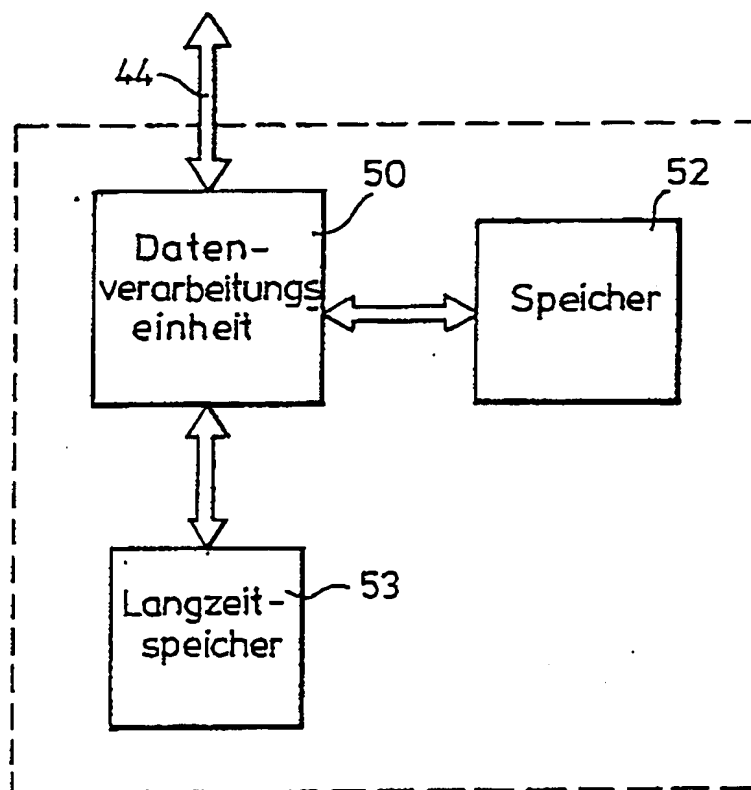


Fig. 3

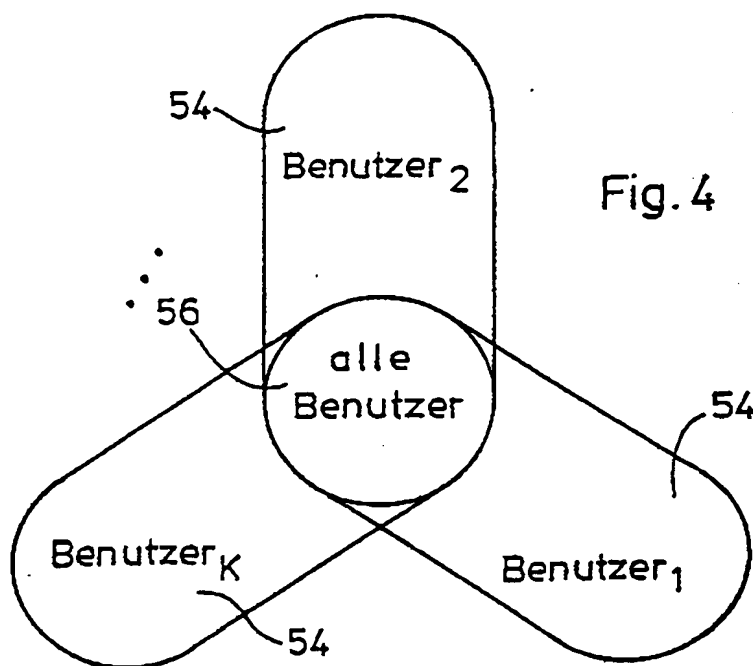


Fig. 4

Fig. 5

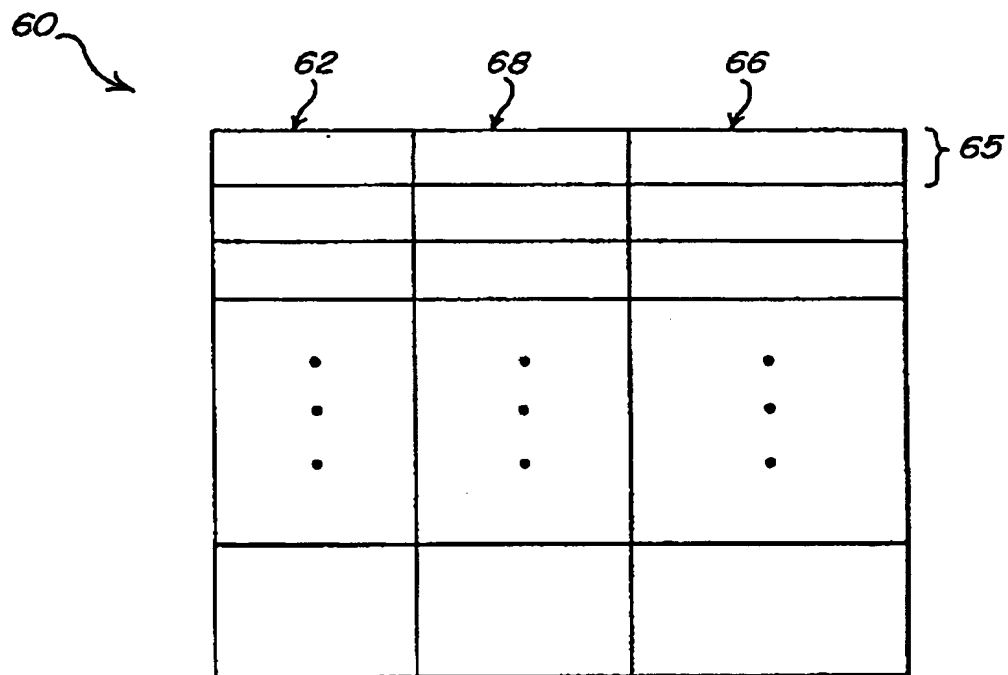


Fig.6

70 →

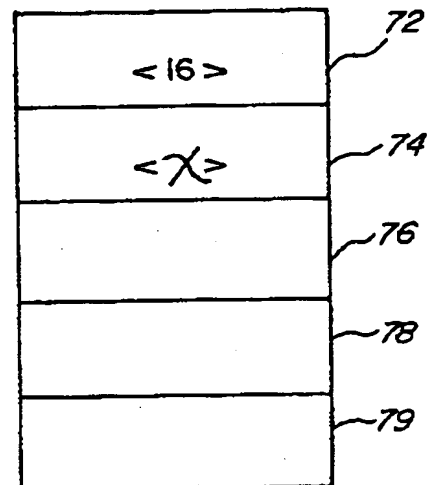


Fig. 7

76 ↘

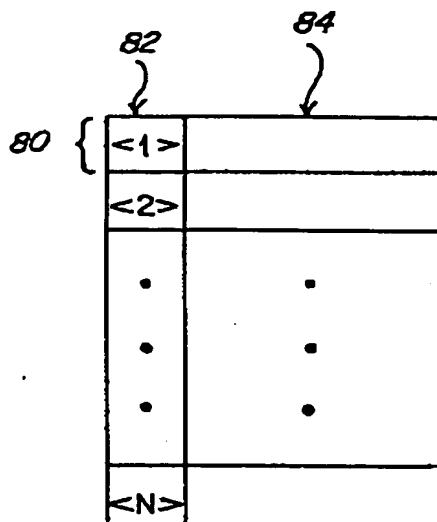


Fig. 8

90 ↘

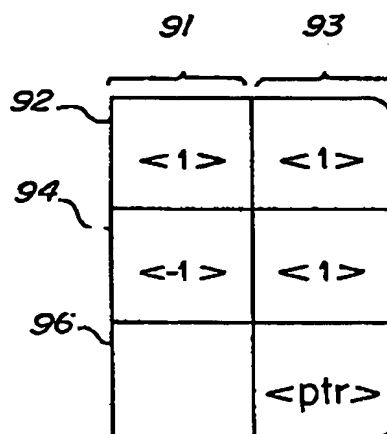


Fig.9

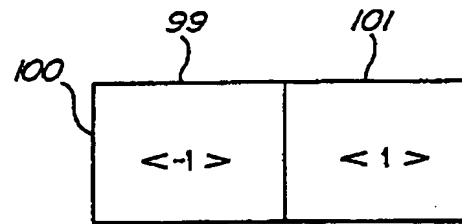


Fig.10

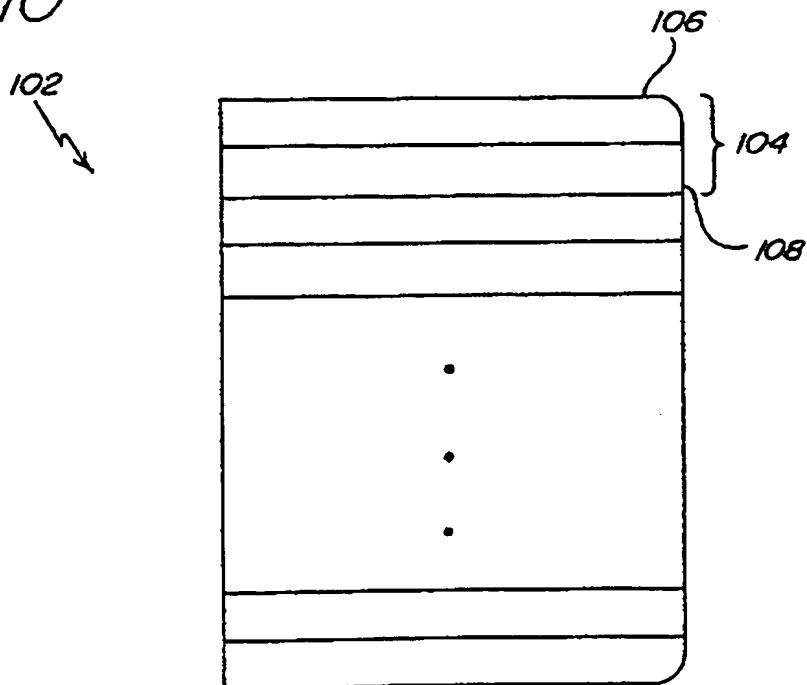


Fig. 11

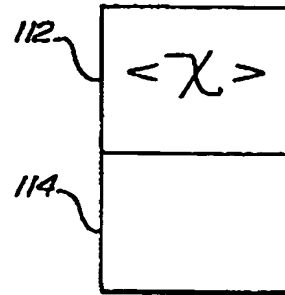


Fig. 12

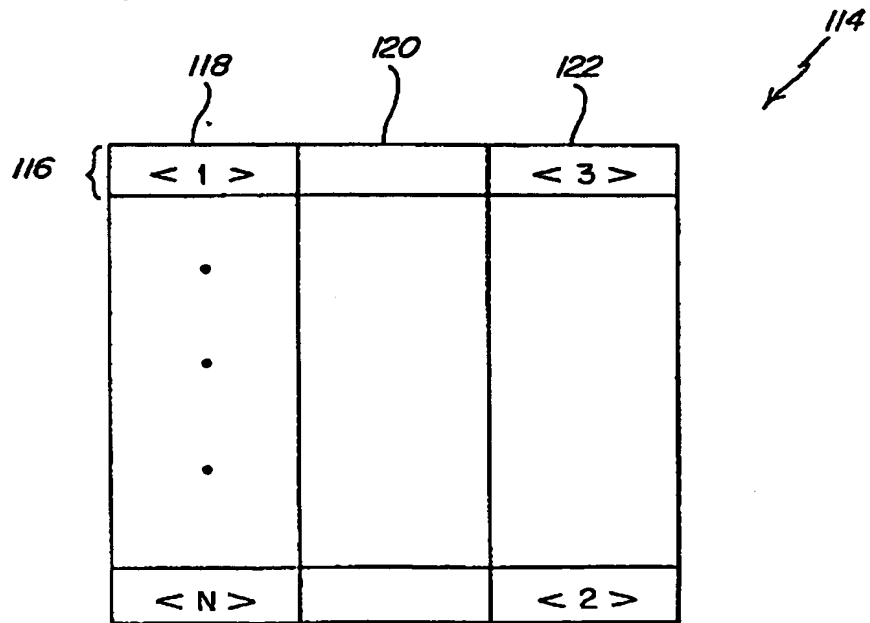


Fig. 13

130 →

	131	133
132	< ptr >	< ptr >
134	< -1 >	< -1 >
136	< -1 >	< -1 >
138	< 0 >	< 4 >
140	< 0 >	< x >
142		< ptr >
144		< ptr >
146		< ptr >

Fig. 14

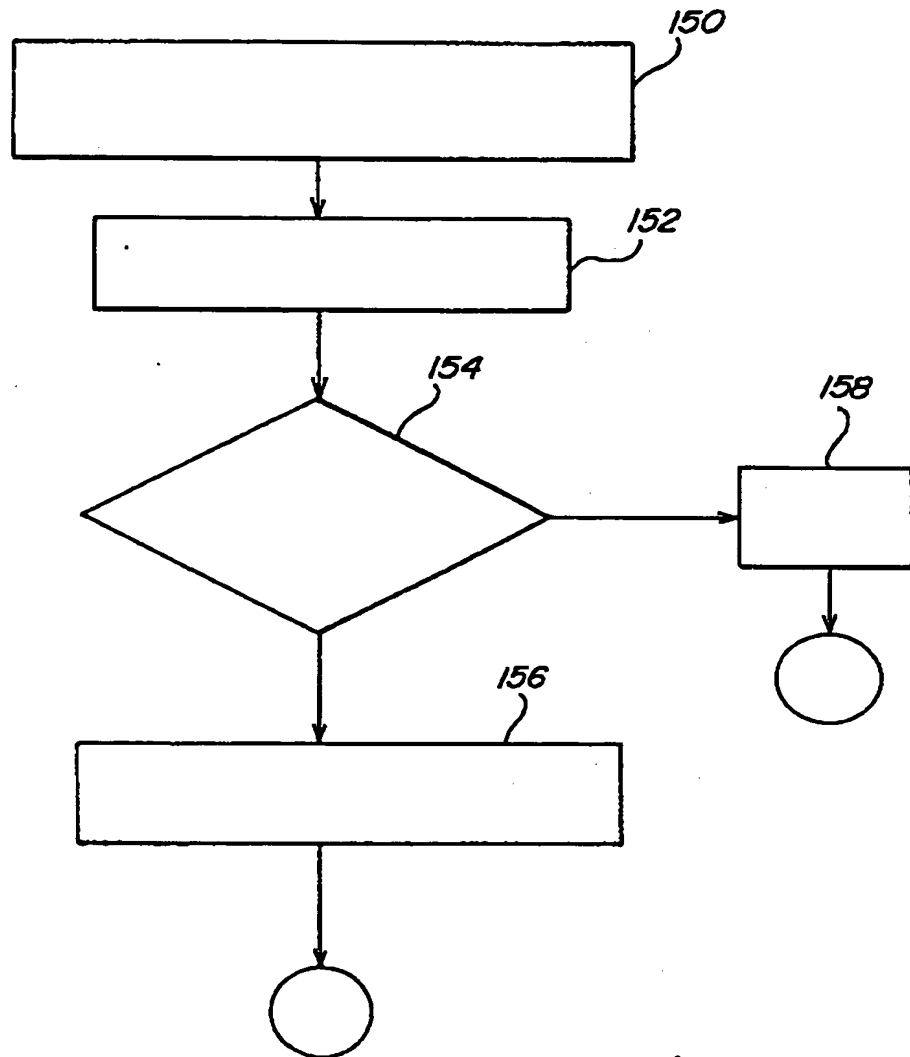


Fig.15

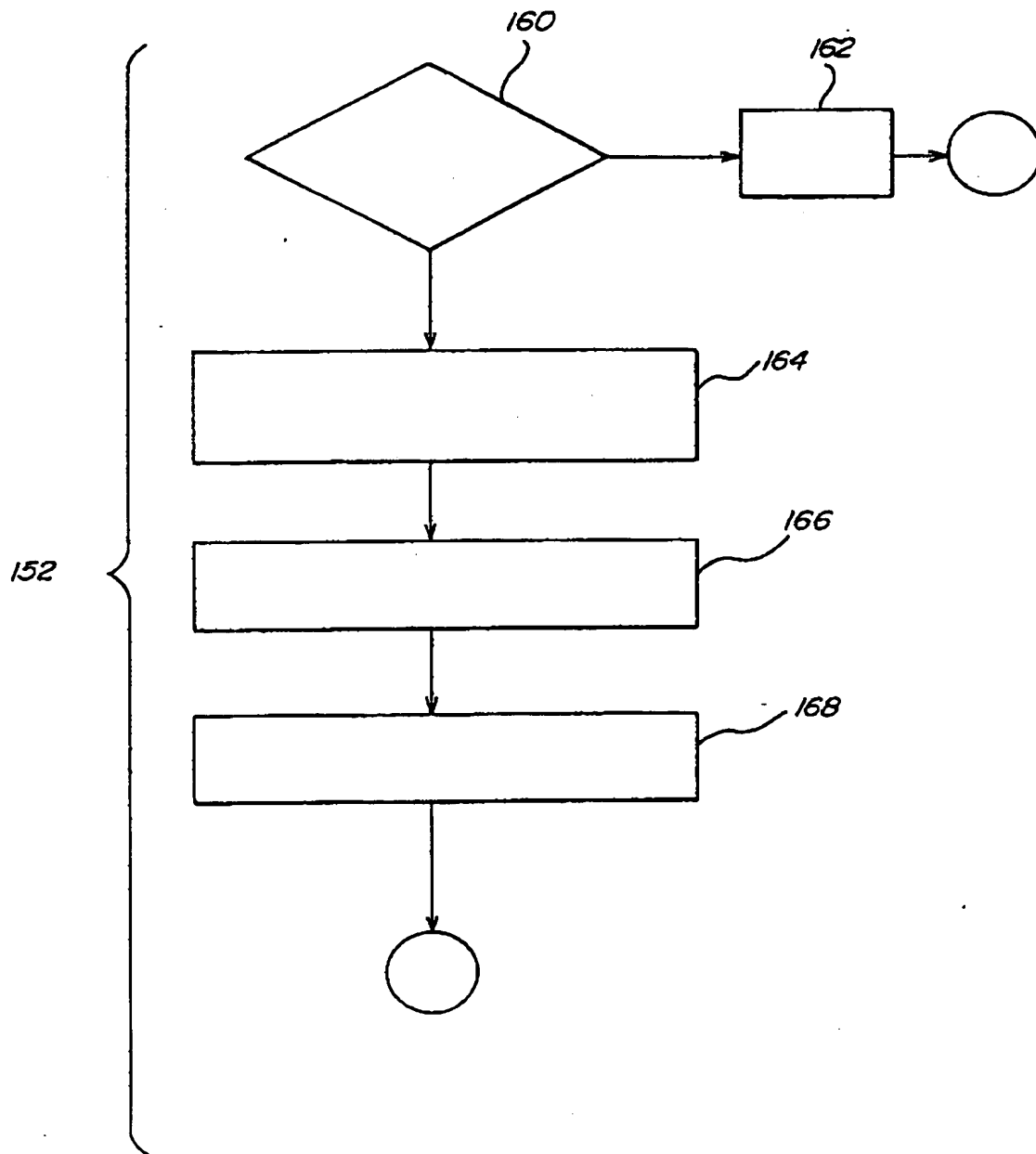


Fig.16

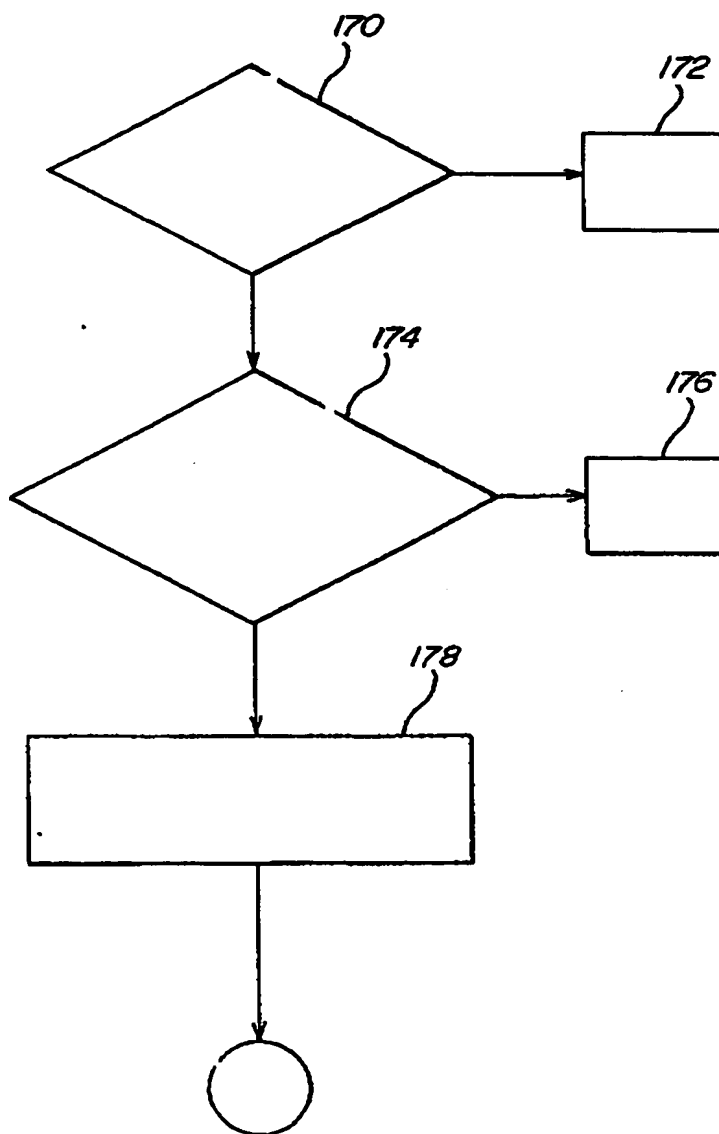


Fig.17

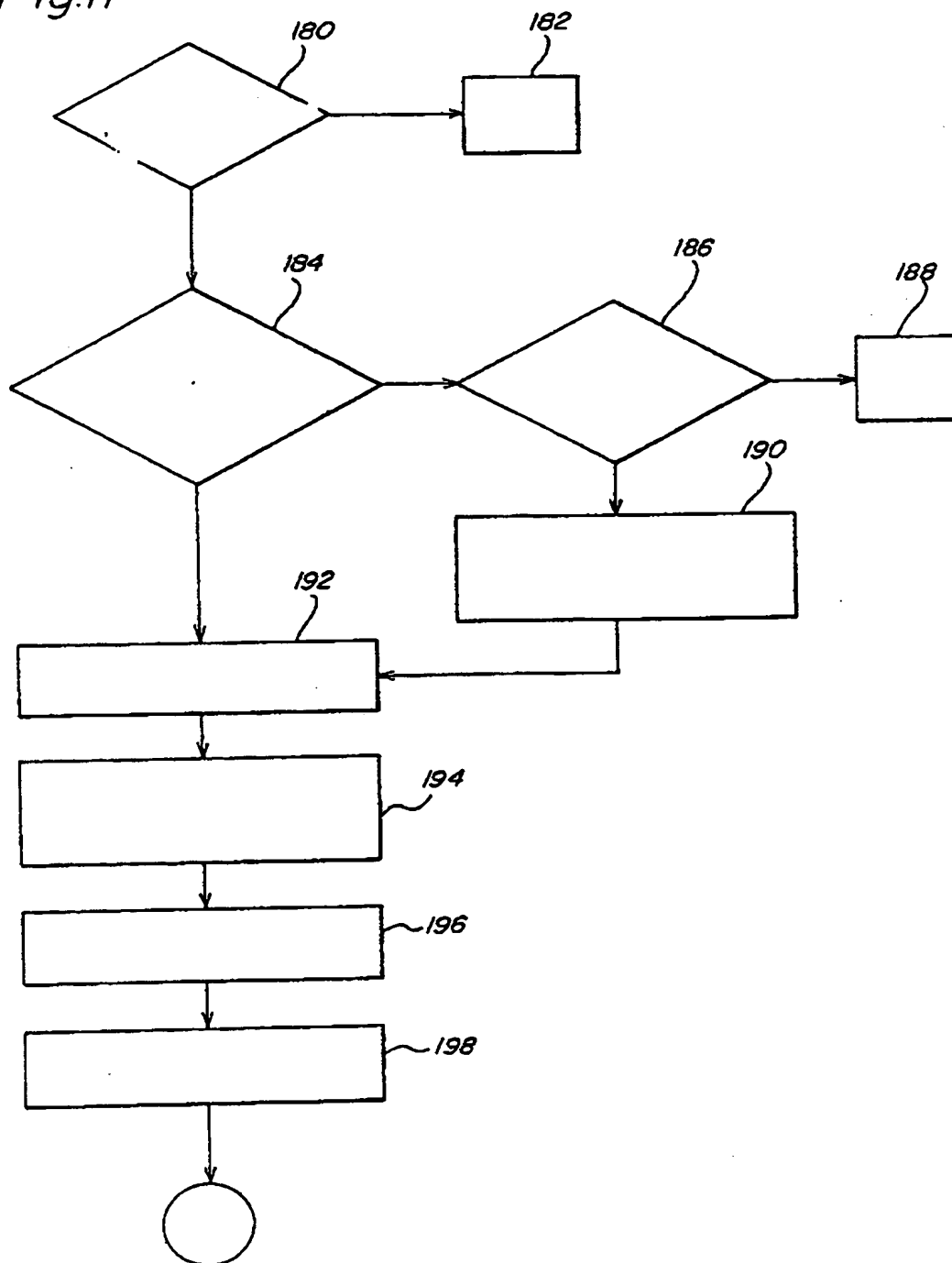


Fig.18

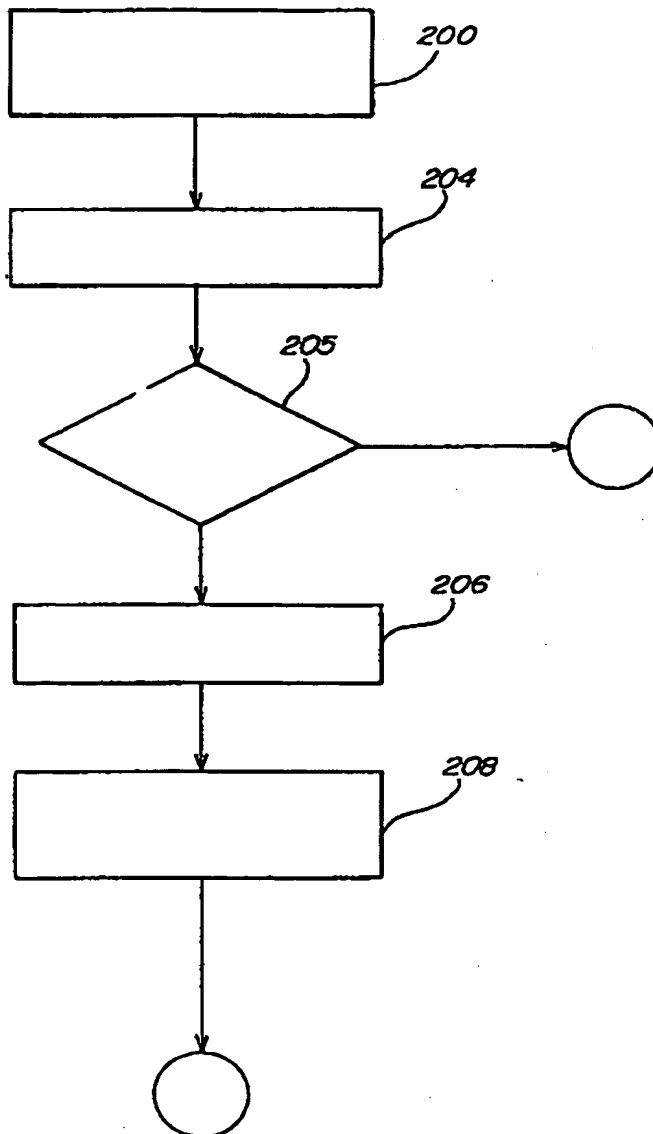


Fig.19

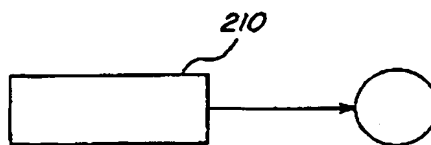


Fig.20

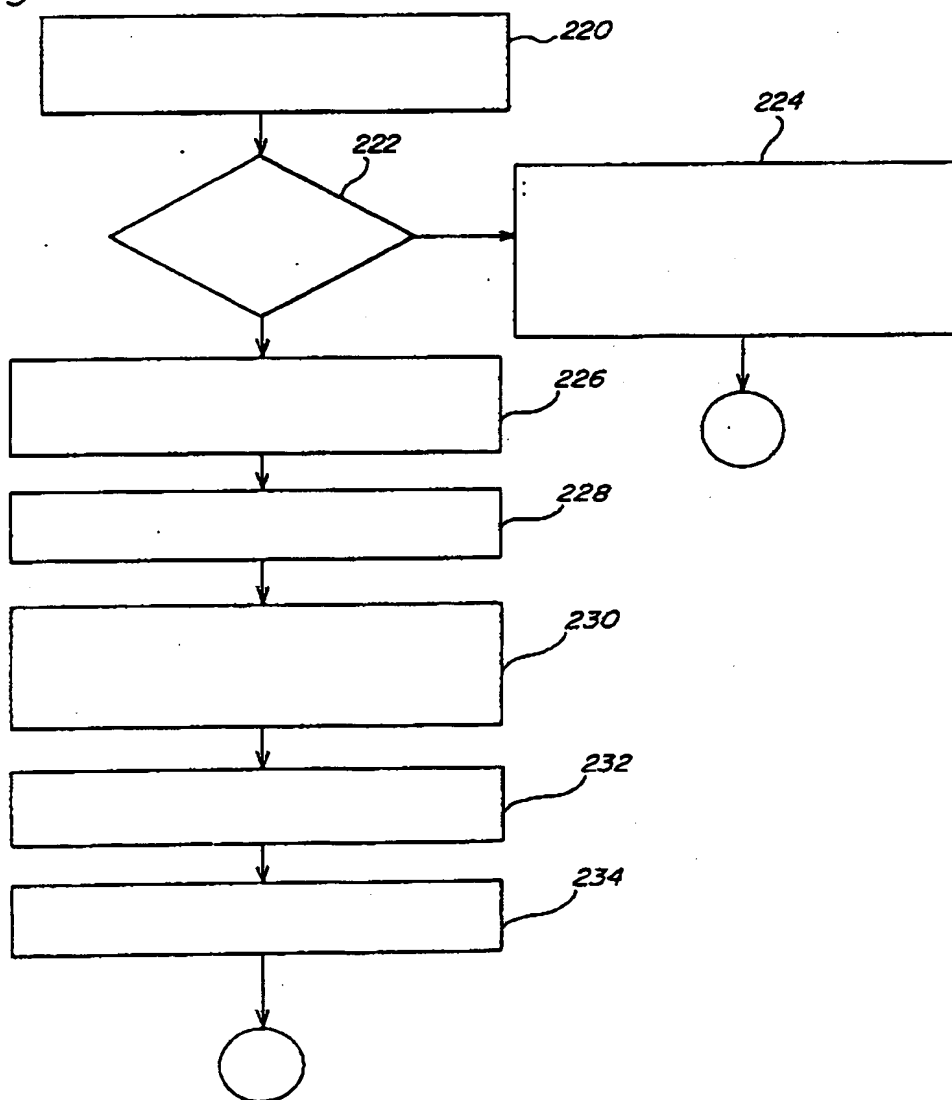


Fig. 21

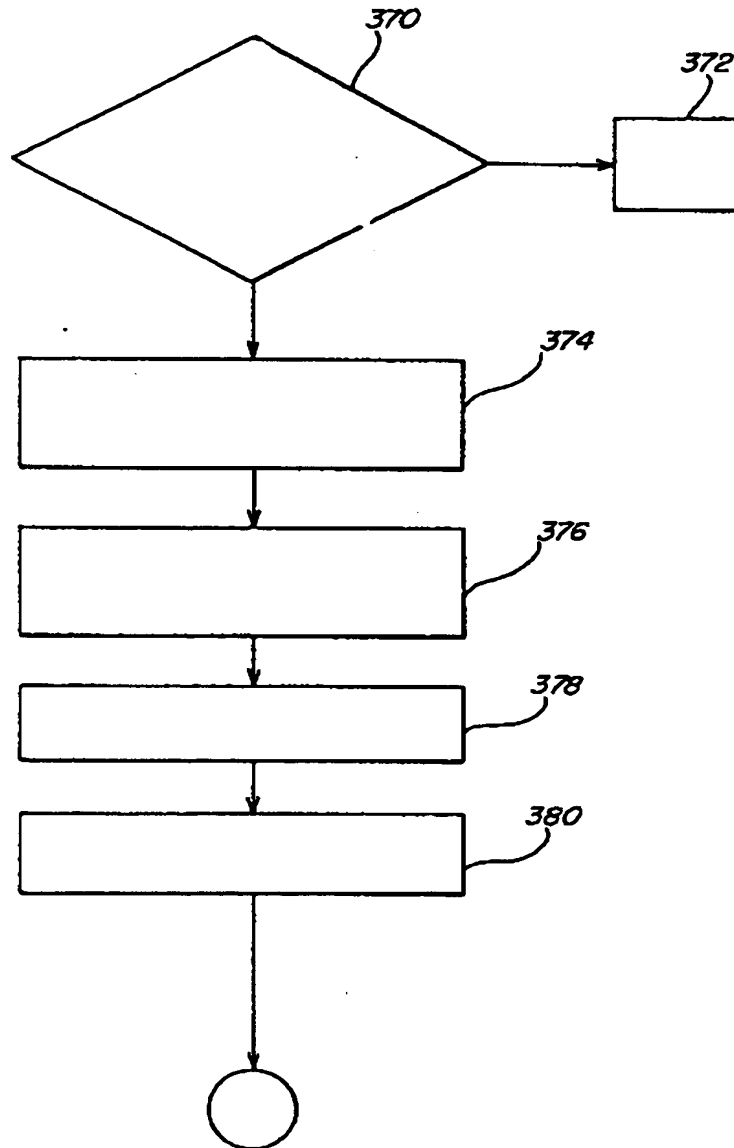


Fig.22

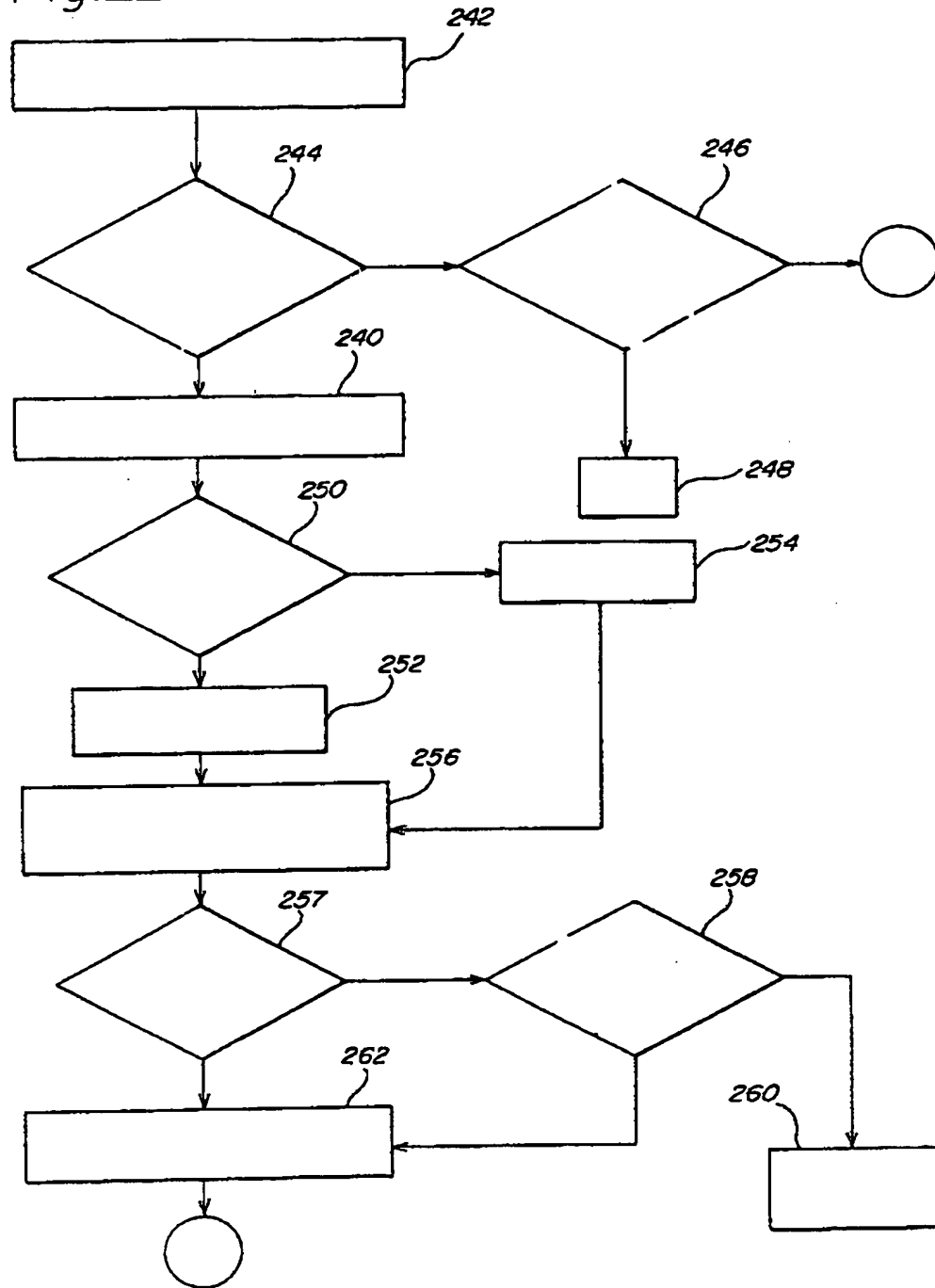


Fig.23A

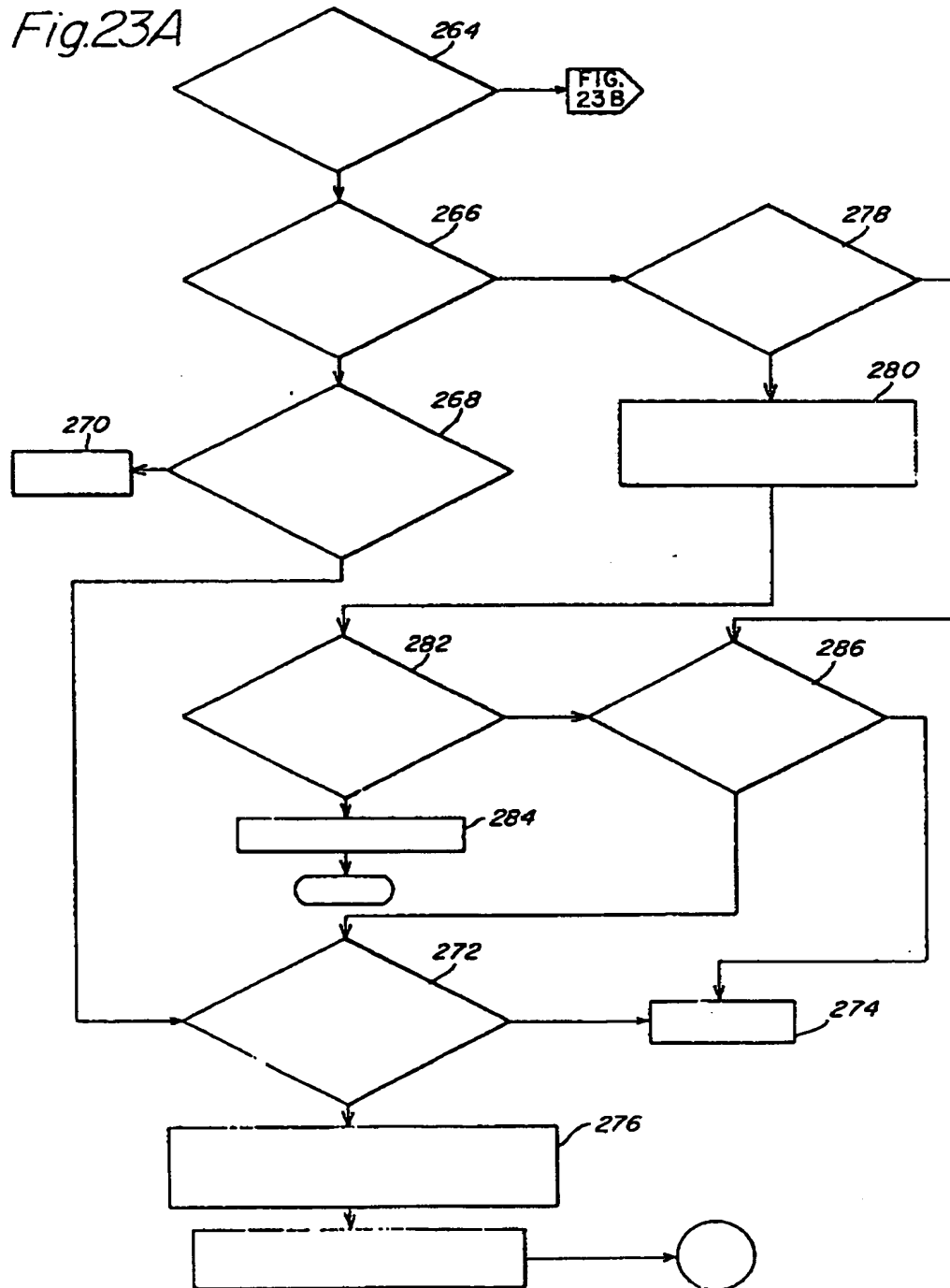


Fig.23B

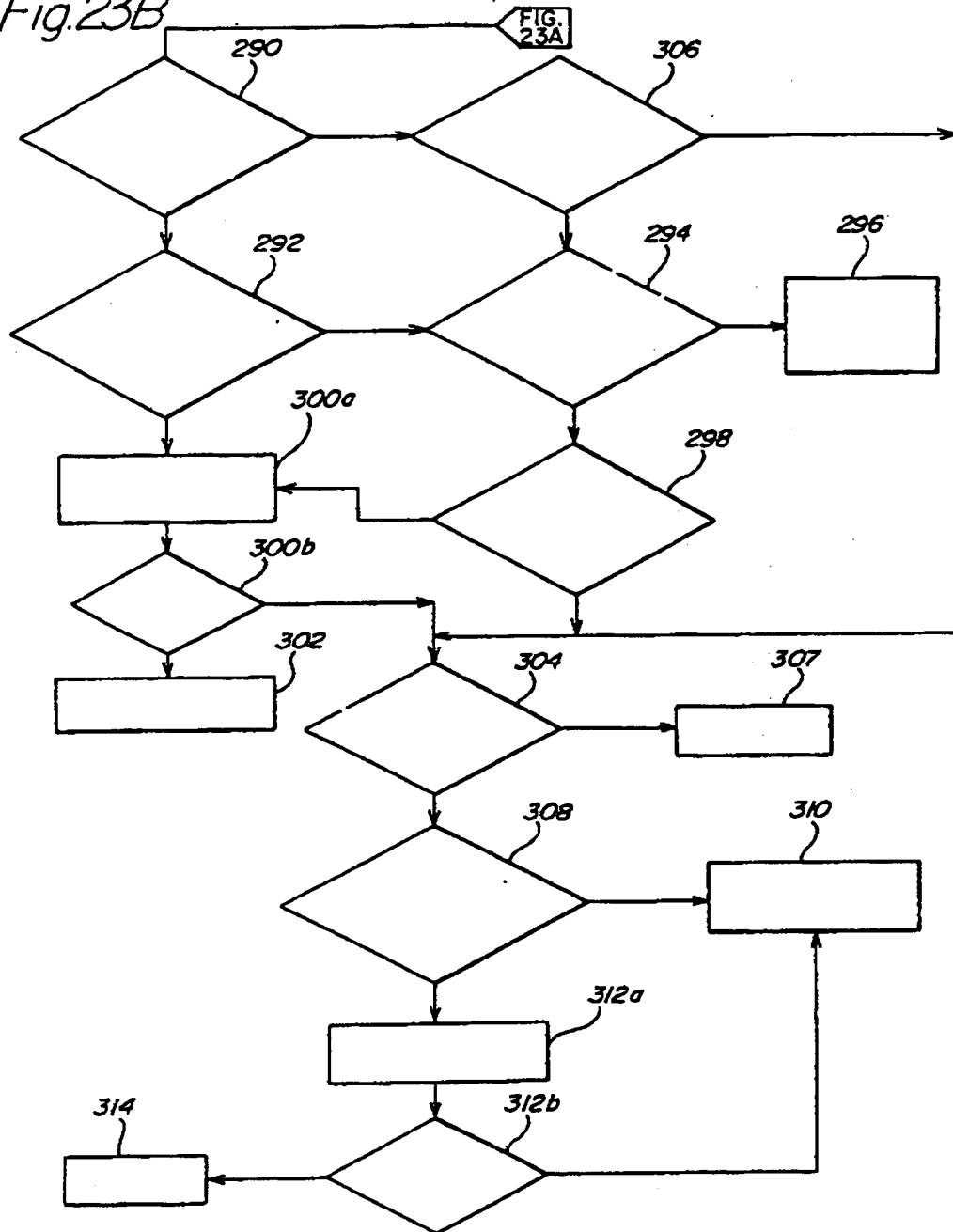


Fig.24

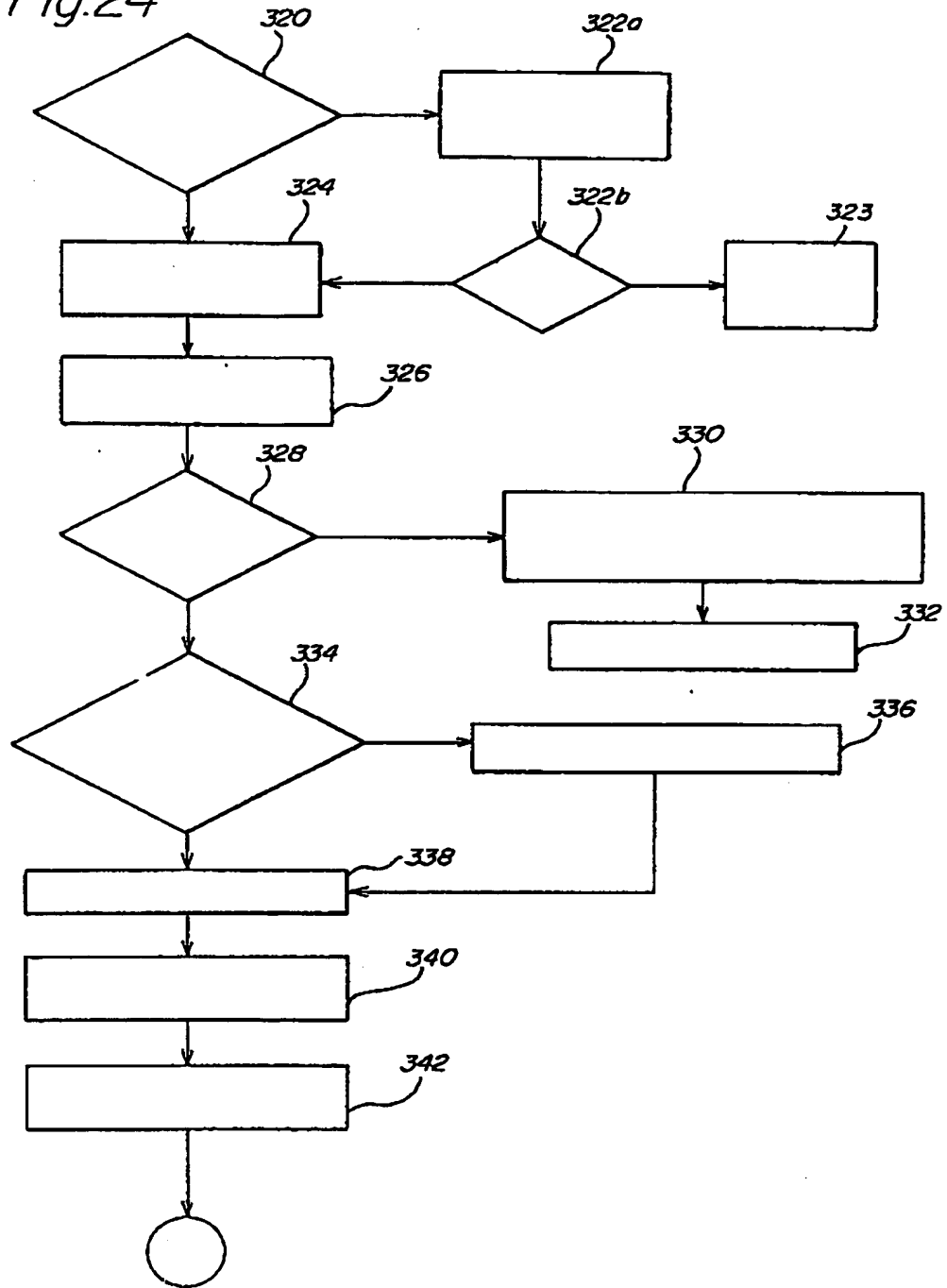


Fig.25

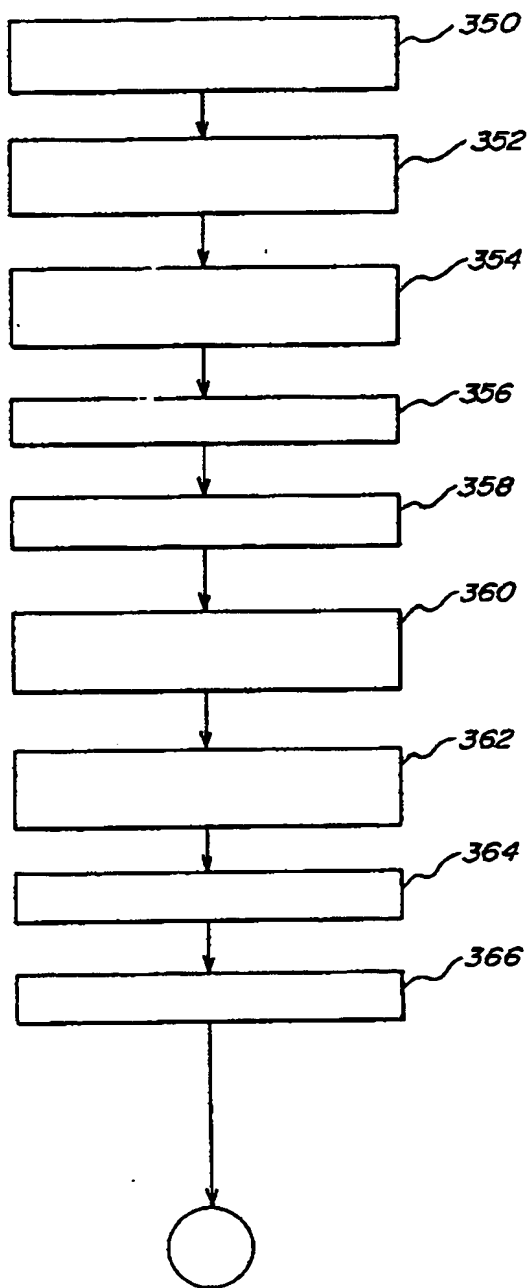


Fig. 26

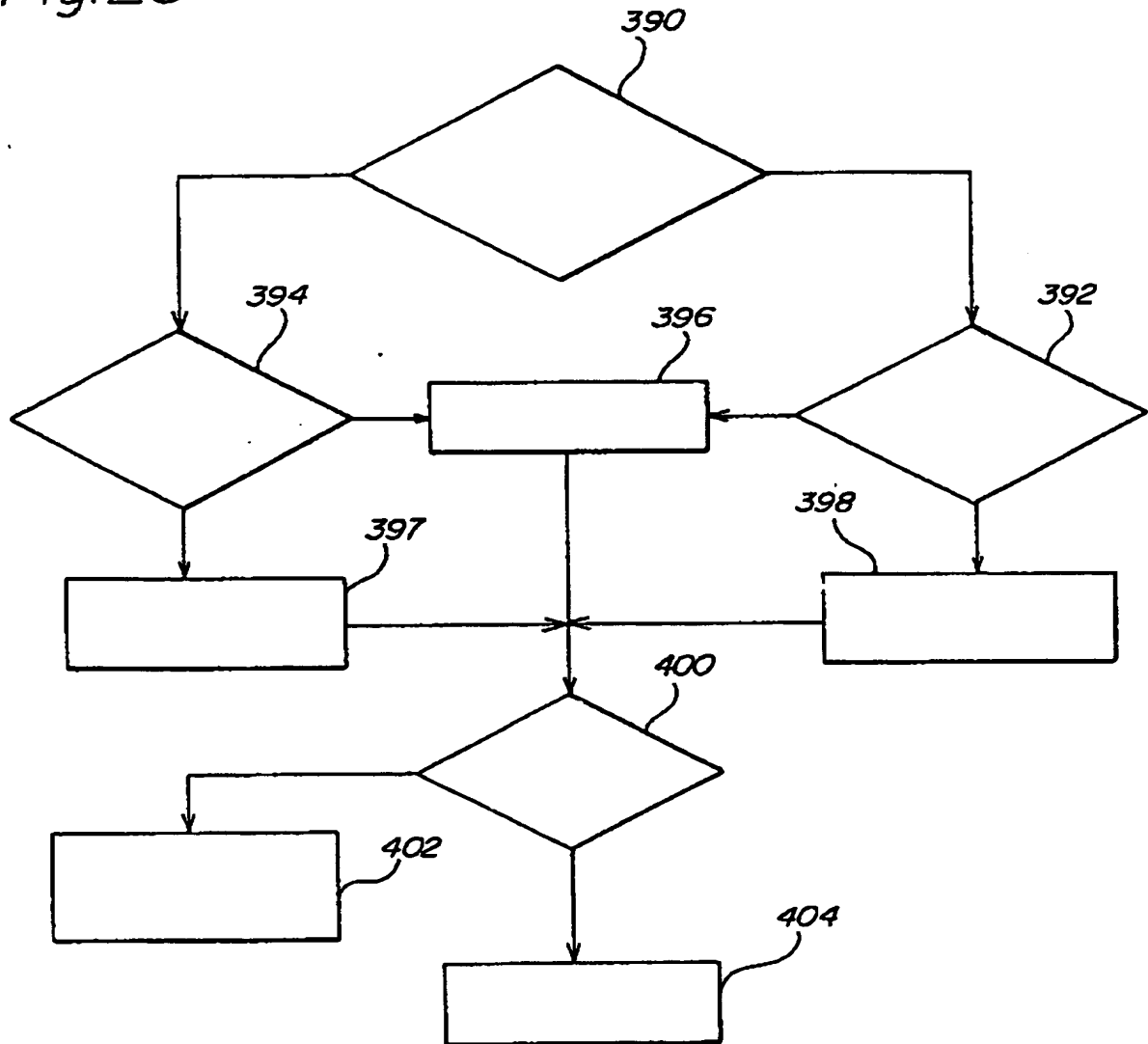


Fig.27

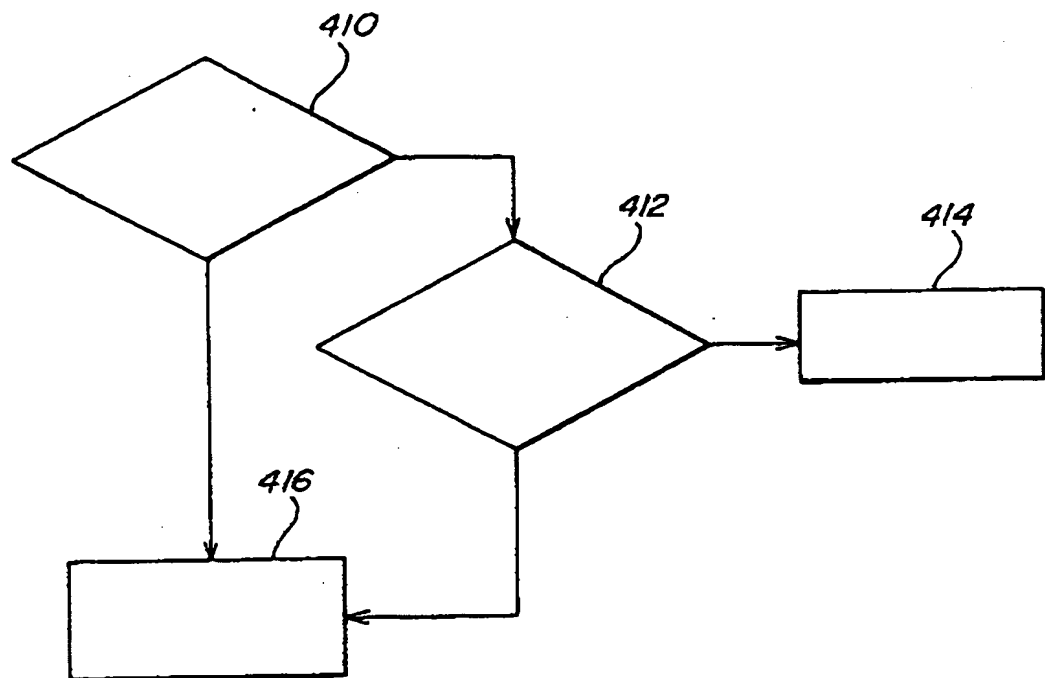
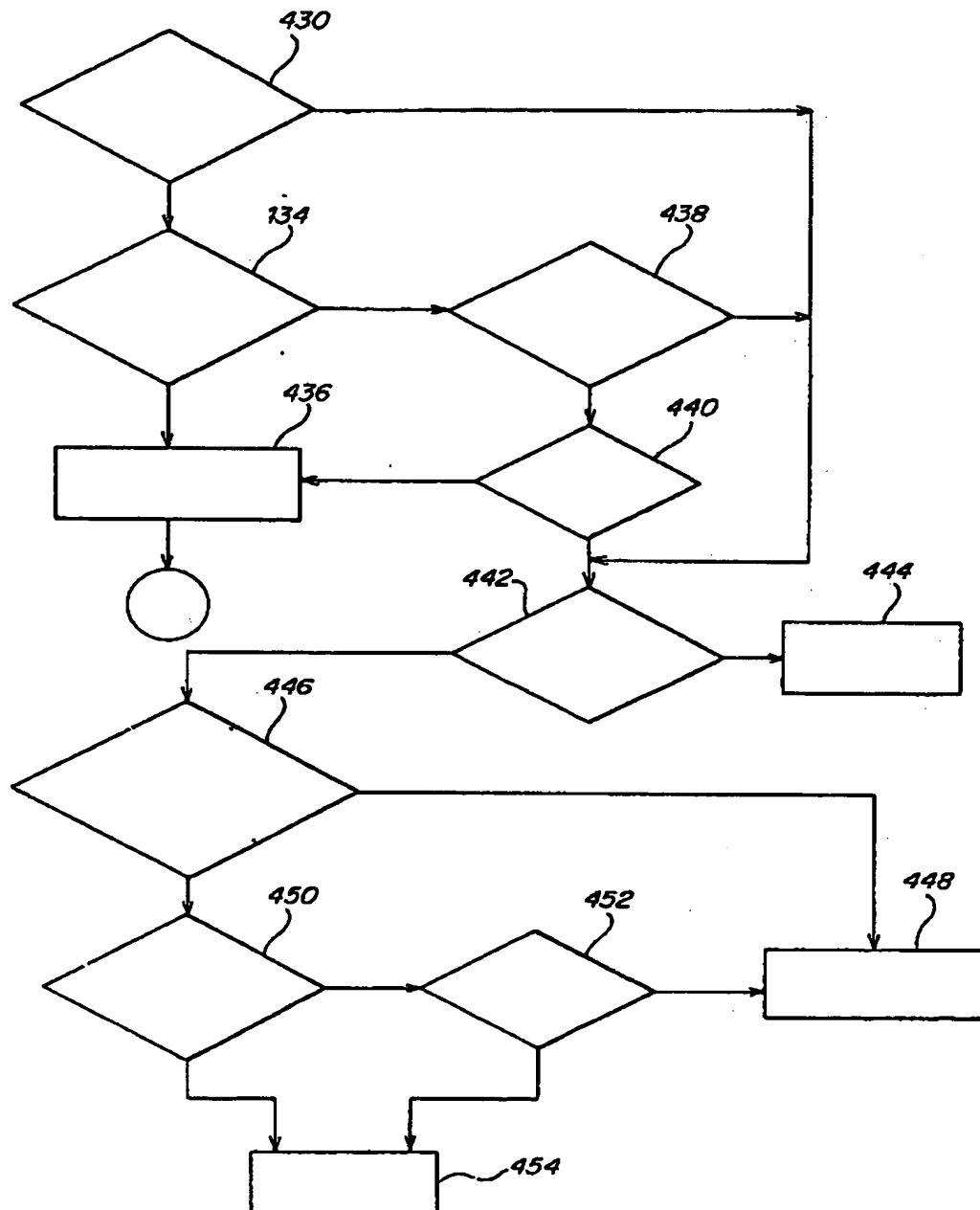


Fig.28



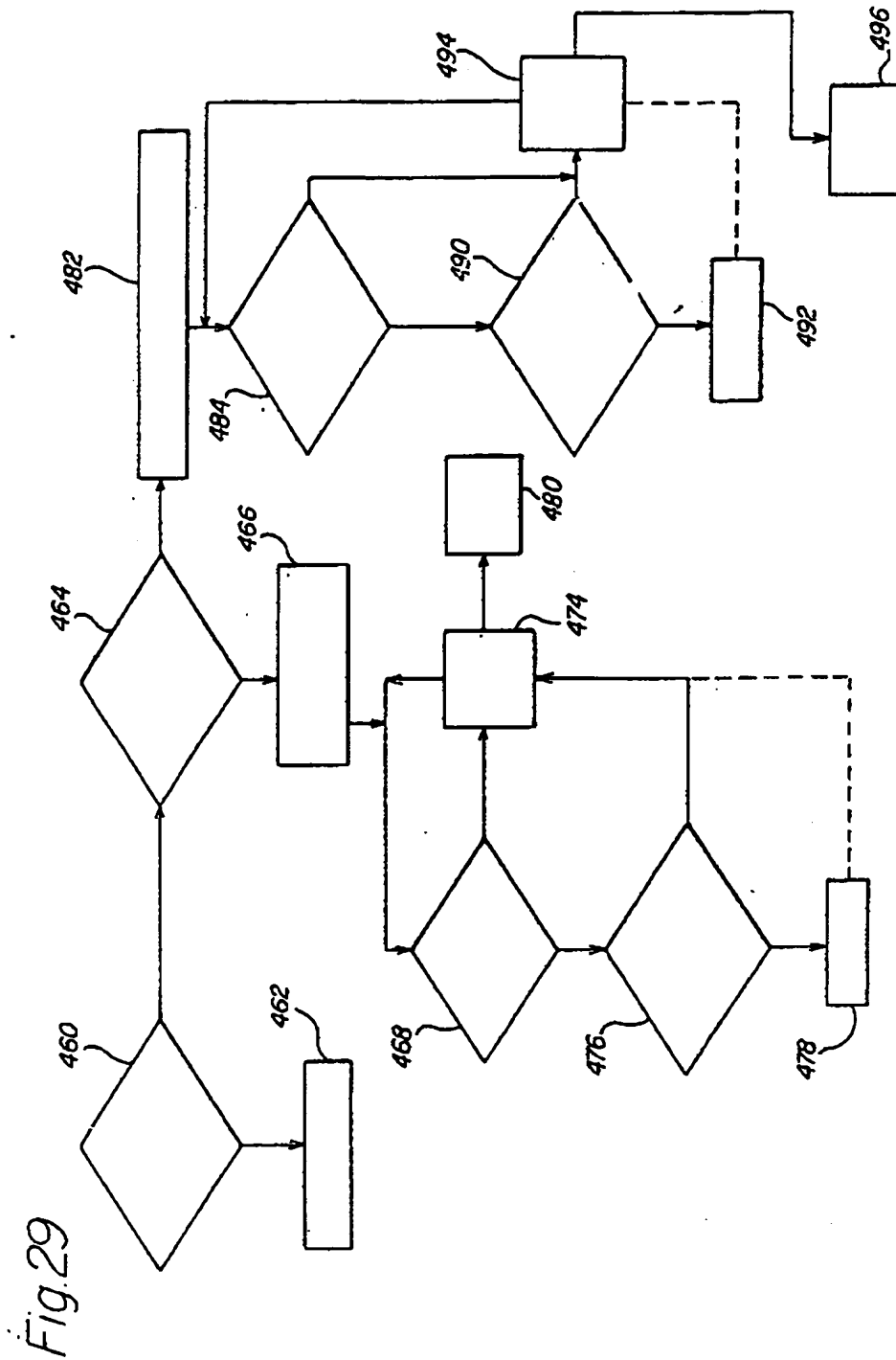


Fig.30

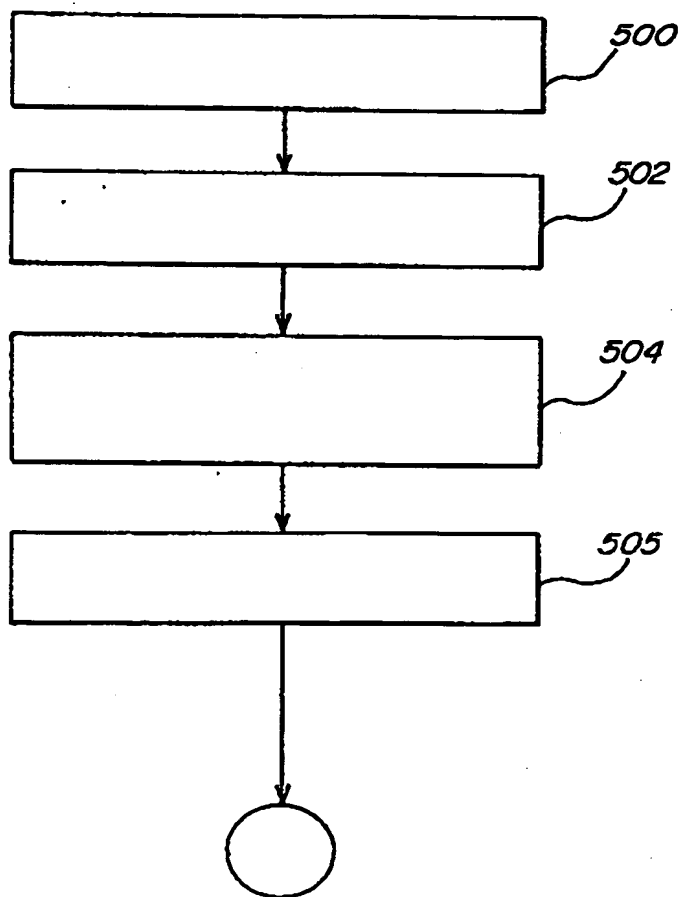


Fig.31

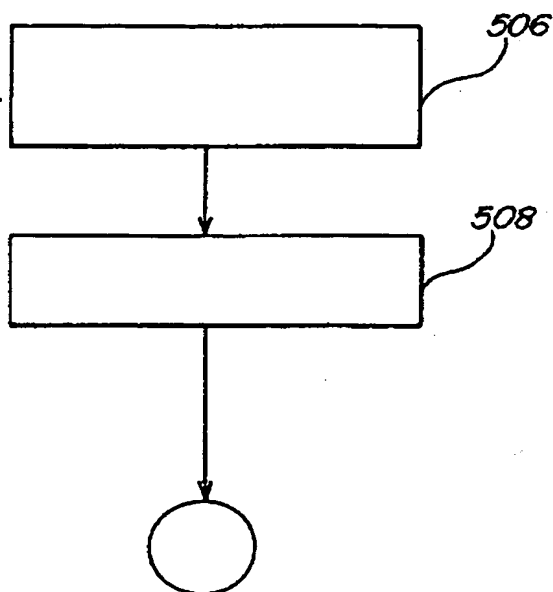


Fig.32

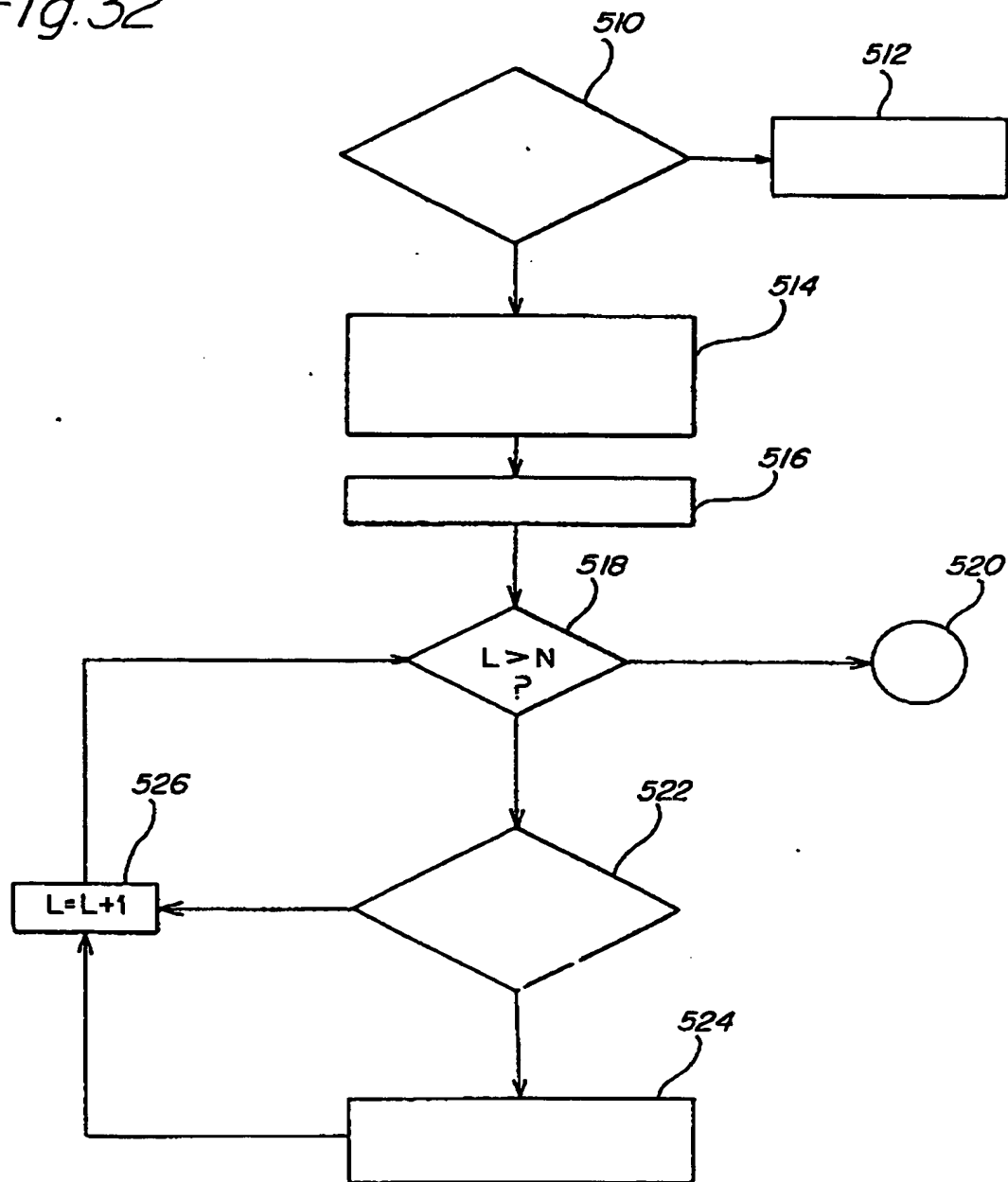


Fig. 33

